

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

EBUDDY TECHNOLOGIES B.V.,	§	
	§	
Plaintiff,	§	
	§	C.A. No.
v.	§	
	§	
LINKEDIN CORPORATION	§	JURY TRIAL DEMANDED
	§	
Defendant.	§	

ORIGINAL COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff, eBuddy Technologies B.V. (hereinafter, “EBT” or “Plaintiff”), by and through its undersigned counsel, files this Original Complaint against Defendant LinkedIn Corporation (hereinafter, “LinkedIn” or “Defendant”), as follows:

NATURE OF THE ACTION

1. This is a patent infringement action to stop Defendant’s infringement of United States Patent Nos. 8,510,395 (the “395 Patent”)(attached as Exhibit 1), 9,584,453 (the “453 Patent”)(attached as Exhibit 2), 8,230,135 (the “135 Patent”)(attached as Exhibit 3) and 8,402,179 (the “179 Patent”)(attached as Exhibit 4) (collectively, the “Patents-in-Suit”).

PARTIES

2. Plaintiff eBuddy Technologies B.V. is a private limited liability company incorporated under the laws of the Netherlands.

3. Upon information and belief, Defendant LinkedIn is a corporation organized and existing under the laws of the State of Delaware, with a place of business at 2029 Stierlin Court, Mountain View, California 94043, and can be served through its registered agent, Corporation Service Company, 251 Little Falls Drive, Wilmington, Delaware 19808.

JURISDICTION AND VENUE

4. This action arises under the Patent Laws of the United States, 35 U.S.C. § 1 et seq., including 35 U.S.C. §§ 271, 281, 283, 284, and 285. This Court has subject matter jurisdiction over this case for patent infringement under 28 U.S.C. §§ 1331 and 1338(a).

5. This Court has personal jurisdiction over LinkedIn, including because LinkedIn is a Delaware corporation; LinkedIn has minimum contacts within the State of Delaware; LinkedIn has purposefully availed itself of the privileges of conducting business in the State of Delaware; LinkedIn regularly conducts business within the State of Delaware; and Plaintiff's cause of action arises directly from LinkedIn's business contacts and other activities in the State of Delaware, including at least by virtue of LinkedIn's infringing methods, systems, computer-readable media, and products, which have been, and are currently, at least practiced, made, and/or used in the State of Delaware. Defendant is subject to this Court's specific and general personal jurisdiction, pursuant to Constitutional Due Process and the Delaware Long Arm Statute. Defendant is subject to this Court's general personal jurisdiction due at least to its continuous and systematic business contacts in Delaware, including related to operations conducted in Delaware and the infringements alleged herein. Further, on information and belief, LinkedIn is subject to this Court's specific jurisdiction, including because LinkedIn has committed patent infringement in the State of Delaware, including as detailed herein. In addition, LinkedIn induces infringement of the Patents-in-Suit by customers and/or infringing users located in Delaware. Further, on information and belief, LinkedIn regularly conducts and/or solicits business, engages in other persistent courses of conduct, and/or derives substantial revenue from goods and services provided to persons and/or entities in Delaware.

6. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391 and 1400(b), including because LinkedIn resides in the State of Delaware at least by virtue of the fact that it is incorporated in this state and at least some of the direct and/or indirect infringement of the Patents-in-Suit occurs in this District.

THE PATENTS-IN-SUIT

7. EBT is the owner of all right, title, and interest in the Patents-in-Suit, including the right to sue for past, present, and future infringement thereof and to collect damages for any such past, present, or future infringement. The inventions disclosed and claimed in the '395 and '453 Patents comprising, *inter alia*, contact aggregation between different messaging services, provide numerous benefits over any prior systems or methods. The inventions disclosed and claimed in the '135 and '179 Patents comprising, *inter alia*, event notification, provide numerous benefits over any prior systems, methods, or non-transitory computer-readable media.

The '395 and '453 Patents

8. The matters described and claimed by the '395 Patent generally include, *inter alia*, systems comprising a network login engine; a network contacts database embodied in one or more non-transitory computer readable mediums; a web server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine: controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider, updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider, maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and

a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device.

9. The matters described and claimed by the '395 Patent also generally include, *inter alia*, methods comprising joining a high level network; joining a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider; obtaining a first contact list associated with the first messaging service provider; obtaining a second contact list associated with the second messaging service provider; maintaining an aggregated contact that comprises the first contact list and the second contact list; logging into the high level network; displaying the aggregated contact list.

10. The asserted claims of the '395 Patent, when viewed as a whole, including as an ordered combination, are not merely the recitation of well-understood, routine, or conventional technologies or components. The claimed inventions were not well-known, routine, or conventional at the time of the invention, including with regard to message services, including web-based messaging services, over ten years ago, and represent specific improvements over the prior art systems and methods.

11. The matters described by the '453 Patent generally include, *inter alia*, systems for contact list aggregation across a plurality of different networks comprising a network interface; a network login engine coupled to the network interface; a network contacts database embodied in one or more non-transitory computer-readable mediums; a server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of

messaging services through a high level network using the network interface to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.

12. The matters described by the '453 Patent also generally include, *inter alia*, methods for contact list aggregation across a plurality of different networks comprising joining a high level network; joining a plurality of low level networks associated with a plurality of messaging services through the high level network; obtaining a first contact list associated with the plurality of low level networks; maintaining a second contact list associated with the high level network; maintaining a contact list associated with the plurality of low level networks for the plurality of messaging services; logging into the high level network; displaying contacts from the plurality of low level networks and the high level network in an aggregated contact list, the contacts retrieved by logging into the high level network.

13. The matters described by the '453 Patent also generally include, *inter alia*, non-transitory computer readable medium comprising executable instructions, the instructions being executable by a processor to perform a method for contact list aggregation across a plurality of different networks, the method comprising: joining a high level network; joining a plurality of low level networks associated with a plurality of messaging services through the high level network; obtaining a first contact list associated with the plurality of low level networks; maintaining a second contact list associated with the high level network; maintaining a contact list associated with the plurality of low level networks for the plurality of messaging services; logging into the

high level network; displaying contacts from the plurality of low level networks and the high level network in an aggregated contact list, the contacts retrieved by logging into the high level network.

14. The asserted claims of the '453 Patent, when viewed as a whole, including as an ordered combination, are not merely the recitation of well-understood, routine, or conventional technologies or components. The claimed inventions were not well-known, routine, or conventional at the time of the invention, including with regard to message services, including web-based messaging services over ten years ago, and represent specific improvements over the prior art systems and methods.

15. At the time of the '395 and '453 patented inventions, including in instant messaging systems, adding contacts to a user's contact list was cumbersome and inefficient. Contacts were typically added by finding a contact in the messaging service, typically followed by sending an invitation request via that specific messaging service so that the recipient could also accept the sender to his/her contact list (or reject the sender so that the sender could not send messages to the recipient). Conventionally, each such list of contacts was contained wholly within the user's account for each specific, separate messaging service used by the particular user. Conventionally, this manual method of aggregating contacts was required for each separate messaging service used by the user. However, even this conventional functionality was limited to cases where both users were already existing users of a particular system. If one of the users was not a member of a particular messaging service, further, cumbersome steps were required that included, *inter alia*, inviting a user to join the messaging service (*e.g.*, via phone call or email invitation) and requiring that user to then sign up and/or create an account for the messaging service.

16. At the time of said inventions, prior art systems and methods were not able to aggregate contacts between different instant messaging services or to view contact information of one instant

messaging service in another instant messaging service. At that time, the most common and practical way to aggregate contacts was by manually typing each name and e-mail address (and/or other contact information) into the user's contact list.

17. The '395 and '435 claimed inventions address technical problems, including that, at the time of the '395 and '435 inventions, aggregation of contacts from a plurality of messaging services was unavailable and/or was cumbersome (e.g., having to manually transfer contact files from one service to another); prone to potential errors (e.g., input errors from manually inputting contact data) and subject to incompatibility issues (e.g., a contact list from one service was not usable or accessible by another service).

18. The '395 and '435 claimed inventions provide specific technological solutions, including permitting, allowing and/or facilitating logging in to first, second or plurality of networks associated with a first, second, and/or plurality of messaging services; obtaining and/or accessing contact information from the first, second and/or plurality of low level networks associated with the first, second and/or plurality of messaging services and aggregating the contacts associated with the first, second and/or plurality of low level networks associated with the first, second and/or plurality of messaging services with the contact from and/or associated with the high level network.

19. The '395 and '435 claimed inventions provide additional specific technological solutions including allowing and/or facilitating logging into a plurality of low level networks to obtain a contact list from the plurality of low level networks and aggregating the contact lists from the plurality of low level networks, including aggregating the contact lists from the plurality of low level networks with contacts from the high level network.

20. The '395 and '435 claimed inventions achieve their inventive improvement in unconventional ways. Without limitation, it was unconventional at the time of the '395 and '435 inventions to log in to and/or facilitate logging in to a plurality of low level networks from a high level network to, *inter alia*, obtain contact lists from the plurality of low level networks and aggregate the contact lists from the low level networks via a contact aggregation engine of the high level network; and it was unconventional to have a network based, application neutral content application, as opposed to the conventional local application specific contacts lists that – if the contacts could be ported at all - had to be manually aggregated via a file exported from one local application and then imported into another local application.

21. Other systems and methods less efficient and more cumbersome than those claimed by the '395 and '453 Patents included exporting a contact list from one application or device as a file, and then importing the file on another application or device. However, this was inefficient and cumbersome, and it required multiple actions by the user. For example, U.S. Patent No. 7,469,379 to Kaplan ("Kaplan"), which was cited and addressed extensively during prosecution of both the '395 and '453 Patents, describes an inferior, prior art system wherein a wireless network maintains a message contact list on a network server and makes the contact list available to devices that can connect to a radio frequency ("RF") card serviced by a single messaging service provider. The single contact list of Kaplan could be provided to different devices to which the RF card was connected. However, Kaplan did not describe, *inter alia*, the contact aggregation engine of the claimed inventions, nor did Kaplan describe obtaining a first contact list associated with a first messaging service provider and a second contact list associated with a second messaging service provider or maintaining an aggregated contact list that comprises the first and second contact lists. Further, Kaplan did not describe accessing a plurality of low level networks through a high level

network to access a contact databases, or the plurality of low level networks associated with a plurality of messaging services as claimed in the '395 and '453 Patents.

22. In addition, conventional systems and methods required manual updating, by typing in new contacts, or doing a new export/import with the risk of double and/or inconsistent contacts because the export typically included all contacts again.

23. These prior art methods and systems for aggregating contacts were in use and available; however, they were and still are inferior to the inventions of the '395 and '453 Patents including because, as described herein, they are more cumbersome, error prone and less efficient, and they require more steps. Reasons that prior art systems and methods were inferior include their lack of an inventive contact aggregation engine, their lack of an inventive network login engine associated with a network for logging to other networks which allows, *inter alia*, a user to aggregate contacts from such other networks, their lack of a network, e.g., Internet, based application neutral, content aggregation solution, and/or their lack of an inventive process for logging into low level networks through a high level network. Inventive concepts embodied in the '395 and '453 claimed inventions thus comprise a network login engine for logging into other networks which allows, *inter alia*, a user to aggregate contacts from such other networks; a network, e.g., Internet, based application neutral content aggregation solution as contrasted with state of the art application specific contact lists that could only be aggregated by exporting contacts from a locally run application into a file and then back into a locally run separate application, and/or their inventive process for logging into low level networks through a high level network. Importantly, including as noted herein and in the '395 and '453 Patents, this functionality is not merely an automation or combination of existing and/or generic components that existed in the art, such as exporting one contact list from one messaging service and importing it in a contact list of another messaging

service. In fact, computers were already used in connection with inferior prior art systems and methods, albeit in inferior ways that lacked the inventiveness of the claimed inventions, including those inferior systems and methods described herein and in the '395 and '453 Patents.

24. The technology claimed in the '395 and '453 Patents does not unduly preempt other system and/or methods, including any for aggregating contacts. For example, the claims do not preempt individually adding contacts to a contact database. Nor do the claims preempt exporting a file containing contact data from one system and then importing that file, including the contact data, into a second system. Nor do the claims unduly preempt lesser ways of aggregating contacts that involve less than the inventive sum of the claimed features.

25. The inventions claimed in the '395 and '453 Patents are directed, among other things, to solutions to the foregoing and other problems or shortcomings in the art at the time of the invention. The claimed inventions of the '395 and '453 Patents comprise novel and inventive techniques and methods for contact list aggregation across networks, including those involving logging into low level networks through a high level network. For example, using certain technology claimed in the '395 and '453 Patents, it becomes possible to, among other things, add contacts from one network or service to a contact list maintained at or by another network or service. This provides an inventive, more efficient, less error prone, user friendly way to add or aggregate contact information from multiple messaging services across multiple networks. As opposed to prior art systems, the inventions claimed in the '395 and '453 Patents, among other things, eliminate the requirement that contact details be memorized, typed in, and/or copied one-by-one, which also avoids typographical and other user-based errors. Moreover, contacts are aggregated centrally, so they can be made available on multiple user devices from a single high level network service. Also, they can be automatically updated and synchronized, including without manual user interaction.

Including in this way and other ways, the efficiency, interoperability and accuracy of contact databases is increased, and they may be kept more up to date, and even continuously up to date, with fewer steps, fewer errors, and less effort.

26. The functionality of contact lists and contact information as known at the time of the invention (e.g., exporting and importing files containing contact information or individually adding contact information to an existing contact list) would be implemented on a user device such as by logging in from the user device to a server and updating contact information, which would be performed by the user on the user device. Thus, adding a contact aggregation engine and/or network login engine, as claimed and described in the '395 and '453 Patents, including to the claimed systems and methods as a whole, was not known, conventional, straightforward or generic. In addition to the foregoing, at the time of the inventions of the '395 and '453 Patents, a messaging system assumed that a user (including through a user device) logged in to a single messaging server system. One messaging system logging into one or more other messaging server systems, including to obtain one or more low level contact lists and then maintain an aggregate contact list, is a substantially different approach that was novel and inventive at the time of the invention and required more than the generic application of computers and/or existing technology.

27. Further, a messaging system with a network login engine that logs in or facilitates login to other networks associated with service providers was inventive and not conventional. Further, a contact aggregation engine that controls the network login engine and updates the network contacts database with contact information from the other networks and stores the aggregated contact list in a non-transitory computer readable medium was inventive and not conventional.

28. Among other things, the inventors of the '395 and '453 Patents wanted to make contact information of a user's contacts in a plurality of messaging services available, including in the

inventive claimed ways, in another messaging service. In this way, a user would, *inter alia*, be able to continue using their preferred messaging service while maintaining, viewing and/or accessing contacts from other messaging services.

29. Importantly, including as noted above, this functionality is not merely an automation or combination of existing and/or generic components that existed in the art, such as exporting one contact list from one messaging service and importing it in a contact list of another messaging service. Further, at the time of inventions of the '395 and '453 patents, messaging services typically used proprietary formats and/or protocols which made it difficult to aggregate contacts across multiple platforms. Additionally, at this time, messaging services typically comprised a stand-alone (e.g., not web-based) application on the user's computer.

30. Further advantages and inventive features of the claimed inventions of the '395 and '453 Patents include that the system or method may comprise a user profile database. By storing user-specific information that the network login engine uses to login or facilitate login to the one or more networks, contact information can be retrieved and an up-to-date aggregate contact list can be maintained, with minimum effort by the user. Further, the user-specific information stored or maintained by the system may be used by the network login engine to login or facilitate login to one or more networks, including one or more low-level networks. For example, the user can provide login information once, and the system can login on behalf of the user without further user action to maintain an up-to-date aggregate contact list, including by the system accessing the low level network using the stored information and retrieving the user's contact list for that network to update the user's high level network contact list.

31. Using the inventive technology described and claimed in the '395 and '453 Patents, it becomes possible to add contacts to a contacts database in a first messaging service from another

second messaging service, making it much easier for a user to reach all the user's contacts from the first messaging service (e.g., the user's contacts from multiple messaging service providers are efficiently aggregated into a single contacts database). This provides, among other things, a user-friendly and efficient way to add or aggregate contact information from multiple messaging services. The user typically enters login information for the second messaging provider into the network login engine of the first messaging system and the contact aggregation engine (including in conjunction with the network login engine) aggregates contacts of both systems into a single contacts database. Contact details do not have to be memorized, typed in, or copied one-by-one, which, as noted above, also avoids errors. Moreover, contacts are aggregated centrally, so they can be made available on multiple user devices. Also, contact details may be automatically updated and synchronized, which enables better and more complete contact information (e.g., checking double contacts, merging contact information, checking inconsistent contact details). All of these and the other advantages of the claimed inventions illustrate their inventiveness and superiority over inferior systems and methods in existence at the time of invention.

32. The claims of the '395 and '453 Patents are directed to, among other things, specific improvements in computer networking and database functionality and capabilities. Among other things, the claimed inventions improve functionality of contact database systems. Including as noted in the '395 and '453 Patents, the claimed technologies comprise innovative systems and processes which increase at least ease-of-use, accuracy and efficiency in contact acquisition and management of contact databases. The '395 and '453 Patents thus provided concrete applications, including via use of network login engines, network architecture and integrations, user profiles and/or contacts aggregated from high and low level networks, that improve computer and database technology, including for aggregating contacts across networks. Further, by aggregating the user's

contacts under a single application and/or web service, the user is not required to run multiple applications or access multiple web services simultaneously. This, in turn, reduces the use of the user's computing device and resources thereon.

33. The inventive systems and methods comprise in some instances components and/or functionality including in the form of a network login engine functionality that logs in or facilitates login to networks associated with messaging service providers. Such systems and methods also comprise contact aggregation engine functionality that controls the network login engine and updates the network contacts database with contact information from the other messaging service providers and stores the aggregated contact list in a non-transitory computer readable medium. These and other claimed patented features are novel and inventive technical components and/or functionality and are not abstract ideas or merely combinations of generic components.

34. Including in accordance with the facts noted above, the noted inventive improvements of the claimed inventions over the prior art represent meaningful limitations and/or inventive concepts based upon the state of the art over a decade ago. Further, including in view of specific improvements, including those noted herein, the inventions of the asserted claims, including when such claims are viewed as a whole and in ordered combination, are not routine, well-understood, conventional, generic, existing, commonly used, well known, previously known, typical, and the like over a decade ago, including because, until the inventions of the asserted claims of the '395 and '453 Patents, the claimed inventions were not existing or even considered in the field.

35. Including in accordance with the facts noted above, the asserted claims, including as a whole and where applicable in ordered combination, comprise, *inter alia*, a non-conventional and non-generic arrangement of communications between networks of messaging service providers, including to login in or facilitate login to a network including to aggregate contacts into a contacts

database that is a technical improvement to the communications between the devices and web services, including those improvements noted above.

36. The claims of the '395 Patent are necessarily implemented in a networked computer environment, including to allow, *e.g.*, login or facilitating login to a first network ... and a second network (*see, e.g.*, claim 1); and joining a high level network, joining a first low level network ... and a second low level network (*see, e.g.*, claim 7).

37. The claims of the '453 Patent are also necessarily implemented in a networked computer environment, including to allow, *e.g.*, login or facilitating login to a first network ... and a second network (*see, e.g.*, claim 1); and joining a high level network, joining a first low level network ... and a second low level network (*see, e.g.*, claim 7).

38. The claims of the '453 Patent are further necessarily implemented in a networked computer environment, including as a non-transitory computer readable medium comprising executable instructions, the instructions being executed by a processor including to allow, *e.g.*, joining a high level network, joining a plurality of low level networks and logging into the high level network (*see, e.g.*, claim 12).

39. Including in accordance with the facts noted above, the claimed inventions are necessarily rooted in computer technology, *i.e.*, network login engine and contacts aggregation engine functionality, and comprise improvement over prior technologies in order to overcome problems, including those noted above, specifically arising in the realm of computer networks and database management (including for messaging service providers) including in combination with the aggregation of contacts from multiple messaging service providers. The claimed solutions amount to an inventive concept for resolving particular problems and inefficiencies, including those noted above, including in connection with logging into and/or facilitating the login into a plurality of

messaging service providers to obtain and aggregate contacts between such providers into a single contacts database.

The '135 and '179 Patents

40. The matters described and claimed by the '135 Patent generally include, *inter alia*, methods for receiving information of an event that calls for user notification; generating an event notification for the event; associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; providing the at least one of the plurality of character strings in the title array to a process executed by a processor; providing an alternative title based on the at least one of the plurality of character strings to the process; using the alternative title as a title in association with the process.

41. The matters described and claimed by the '179 Patent generally include, *inter alia*, methods for processing an event that calls for user notification; generating an event notification for the event; storing the event notification in an array; providing the event notification from the array to a process executed by a processor; using the event notification as a title in association with the process; providing an alternative title from the array to the process; and using the alternative title as a title in association with the process.

42. The asserted claims of the '135 and '179 Patents, when viewed as a whole, including as an ordered combination, are not merely the recitation of well-understood, routine, or conventional technologies or components. The claimed inventions were not well-known, routine, or conventional at the time of the invention, over ten years ago, and represent specific improvements over the prior art and prior existing systems and methods.

43. At the time of the invention of the '135 and '179 Patents, event notifications typically involved the use of intrusive and/or distracting pop-up windows or sounds, or the use of intrusive and/or distracting text effects on the titlebar (*e.g.*, blinking text) for providing event notification.

44. The use of text effects, like blinking for example, can provide a notification that there is an event, but it is limited in the amount of information it provides about the event, and further requires the user to check the window and/or application to determine whether there is a relevant event (such as a message from a relevant person).

45. At the time of the '135 and '179 patented inventions, it was not known or conventional to provide an event notification in a titlebar or taskbar using an array, including to provide an alternative title.

46. The use of arrays to store character strings for titles in titlebars or taskbars to provide alternating titles was not known at the time of the invention and is not a straightforward consequence of computerizing any ordinary task of notifying a user of an event. Thus the title array of the claimed inventions of the '135 and '179 Patents enable the use of alternative titles and thereby provides, *inter alia*, more detailed information to the users about the notification. By providing the user more information, in a non-intrusive manner, the user is then able to decide whether to respond to a notification now or later without leaving the application currently in use (*e.g.*, the user is able to give priority to the application providing the notification, or first continue his other activities).

47. Importantly, including as noted above, this functionality is not merely an automation or combination of existing and/or generic components that existed in the art, such as renaming the titlebar or taskbar or merely providing a list of events in an array. For example, it was not generally known at the time of the invention of the '135 and '179 Patents to rename the titlebar or taskbar in

a messaging service application to provide additional information and/or draw the user's attention to a particular event.

48. The claims of the '135 and '179 Patents are necessarily implemented by a computer or other computing device, including at least, for example, because the claims require a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device (*see, e.g.,* '135; claim 1 and '179; claim 1) including via a user's web browser, which is necessarily an application run on a computing device.

49. An advantage of notifying a user of an event in a titlebar or taskbar (including as opposed to using a sound or pop-up window) is that it provides a subtle notification to the user, *e.g.,* no frequent and/or intrusive ping-sound or pop-up windows. As noted in the specification, "a user at work may not want to play a noise or have a popup window show up every time a message is received." '135/5:31-32. The user may view the notification in the titlebar or taskbar without having to stop his other activities, for example, having to close pop-up windows. Another advantage of notifying a user of an event in a titlebar or taskbar, including at the time of the invention of the '135 and '179 Patents, was that not all computers included speakers (*e.g.,* the use of a sound notification may not have been available to all users). Further, the use of pop-up windows at this time often required a technical and/or complicated configuration change in the browser itself.

50. One advantage of using a title array is that it reduces some display limitations associated with the titlebar or taskbar limitations. For example, text effects can be displayed and viewed by scrolling through the title array. Further, more information can be displayed by cycling through the title array over time. For example, if the user received messages from multiple senders, the names of the senders may be provided in the array and scrolled through the titlebar or taskbar.

Thus the previous event notification is not lost at the client when a new event notification arrives, possibly without the user noticing the previous event notification, and/or new event notifications do not have to be maintained by the server. And at the end of the array, the title generator may start over at the beginning of the array to continue to display the event notifications (*e.g.*, names of senders). In this way, the names of the senders of unread messages, for example, may be provided repetitively and/or continuously, making it less needed for the user to watch or monitor the titlebar or taskbar all the time, thus further improving the non-intrusive and informative event notification. Moreover, the use of the titlebar display avoids limitations in computing devices which lack sound or other methods of notification used in the prior art.

51. Thus the title array enables the alternative titles and may provide more detailed information to a user about the event notification. By providing the user more information, in a non-intrusive manner the user benefits from receiving additional information but without distraction and without having to leave the application the user is currently using. This may allow a user, for example, to decide whether she or he wants to respond to a notification now or later without leaving the application she or he is using (*e.g.*, give priority to the application providing the notification, or first continue his or her other activities).

52. A meaningful and inventive feature and limitation is in particular the title array, in that the event notification is associated with at least one of the plurality of character strings in a title array that includes a plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device, providing the at least one of the plurality of character strings in the title array to a process executed by a processor, providing an alternative title based on the at least one of the plurality of character strings to the process. Inventive aspects of the claimed inventions comprise providing improved user

notification methods making it easier and more efficient for users to visualize and navigate through multiple notifications; and providing for more efficient, uninterrupted and less signal-strength dependent handling of notifications locally by the browser.

53. The '135 and '179 claimed inventions provide specific technological solutions, including, without limitation, providing less intrusive and/or more informative mechanisms and/or methods for providing event notifications to a user, providing improved user notification methods making it easier and more efficient for users to visualize and navigate through multiple notifications; providing for more efficient, uninterrupted and less signal-strength dependent handling of notifications locally by the browser; including, *inter alia*, by providing an array title array comprising plurality of character strings for display in a titlebar or taskbar of a display device.

54. The '135 and '179 claimed inventions provide specific technological improvements including, without limitation, associating an event notification with at least one of a plurality of character strings in a title array for provisioning for display in the titlebar or taskbar of a display device and further by providing an alternate title from the title array based on the plurality of character strings and using the alternate title as a title, including as described in claim 1.

55. The '135 and '179 claimed inventions achieve their inventive improvements in unconventional ways. Without limitation, it was unconventional at the time of the '135 and '179 inventions to associate an event notification with at least one of a plurality of character strings in a title array for provisioning for display in the titlebar or taskbar of a display device and providing an alternate title based on the at least one of the plurality of character strings.

56. In addition, methods of event notification in browsers that did not use intrusive or disruptive sounds or pop-ups, required the user to periodically switch to a different window and/or to a different program on the computer in order to check if new notifications had been received.

Further, sounds and popups were not always available in every browser and for every user, including the particular browser in use by and/or available for use by a particular user, at the time of the claimed inventions. Unconventional aspects of the claimed inventions comprise using the title bar and using the local browser for event notifications.

57. Other more distracting and intrusive and/or less informative methods of event notification at the time of the '135 and '179 patented inventions included an audible sound (if the user's computer was configured to have sound capability) to indicate, for example, an inbound chat message. However, this was distracting and intrusive to the user (and others who may be working with or around the user) who may be working, for example, in a different window at the time and/or unusable for those with computing devices with the sound off or entirely lacking sound output. For example, U.S. Publication No. 2004/0015547 (Griffin, et al.) describes a buddy list display, including providing a title bar region allowing the display of text and graphic symbols that may be accompanied by an audible sound. However, Griffin did not disclose the use of an array. Nor does Griffin describe providing an alternative title from an array.

58. These prior art methods and systems for event notifications are inferior to the inventions of the '135 and '179 Patents, including because, as described herein, they are more intrusive, limited, and less widely compatible. Reasons that prior art systems and methods were inferior include their lack of cross-compatibility for devices lacking sound output and the intrusive and/or frequent nature of the notifications, their lack of use of alternative titles which allows, *inter alia*, more detailed information to the users about the notification. Importantly, including as noted herein and in the '135 and '179 Patents, this functionality is not merely an automation or combination of existing and/or generic components that existed in the art, such as renaming the titlebar or taskbar or merely providing a list of events in an array. In fact, computers were already

used in connection with inferior prior art systems and methods, albeit in inferior ways that lacked the inventiveness of the claimed inventions, including those inferior systems and methods described herein and in the '135 and '179 Patents, including because they necessitated the use of browsers, titlebars, taskbars, and/or sound output which necessarily exist and function only with the use of a computer.

59. The technology claimed in the '135 and '179 patents does not unduly preempt other system and/or methods, including any for event notification. For example, the claims do not preempt the use of a pop-up window to notify the user of an event. Nor do the claims preempt the use of a sound or flashing text to notify the user of an event. Further, the claims do not preempt the use of any other more intrusive or distracting methods of event notification known in the art at the time of the invention. Nor do the claims unduly preempt lesser ways of event notification that involve less than the inventive sum of the claimed features.

60. The claims of the '135 and '179 Patents are directed to, among other things, specific improvements in computer messaging and notification capabilities. Among other things, the claimed inventions improve functionality of messaging systems. Including as noted in the '135 and '179 Patents, the claimed technologies comprise innovative systems and processes which increase at least ease-of-use, accuracy and efficiency in messaging systems, including such systems comprising event notification functionality. The '135 and '179 Patents thus provided concrete applications, including via use of event notifications, titlebars/taskbars, and title arrays, that improve computer and database technology, including for providing event notifications.

61. The inventive claimed systems and methods of the '135 and '179 Patents comprise in some instances components and/or functionality including in the form of event processing, including generating an event notification and further including providing and/or generating a title array

comprising such event notification. Such systems and methods also comprise providing or provisioning a titlebar or taskbar with the event notification in the title array and further providing or provisions an alternative title from title array. These and other claimed patented features are novel and inventive technical components and/or functionality and are not abstract ideas or merely combinations of generic components.

62. Including in accordance with the facts noted above, the noted inventive improvements of the claimed inventions over the prior art represent meaningful limitations and/or inventive concepts based upon the state of the art over almost a decade ago. Further, including in view of specific improvements, including those noted herein, the inventions of the asserted claims, when such claims are viewed as a whole and in ordered combination, are not routine, well-understood, conventional, generic, existing, commonly used, well known, previously known, typical, and the like over almost a decade ago, including because, until inventions of the asserted claims of the ‘135 and ‘179 Patents, the claimed inventions were not existing or even considered in the field.

63. Including in accordance with the facts noted above, the asserted claims of the ‘135 and ‘179 Patents, including as a whole and where applicable in ordered combination, comprise, *inter alia*, a non-conventional and non-generic arrangement of event notification including by providing or storing an event notification in an array to be provided to and/or displayed in a taskbar or titlebar and further including an alternative title to be provided and/or displayed in a taskbar or titlebar.

64. Including in accordance with the facts noted above, the claimed inventions are necessarily rooted in computer technology, i.e., receiving and/or processing an event notification and associating or storing the event notification in a title array to be provided to and/or displayed in a titlebar or taskbar. The claimed solutions amount to an inventive concept for resolving the

particular problems and inefficiencies noted above, including in connection with providing event notifications to a user in a titlebar or taskbar through the use of, *inter alia*, a title array.

COUNT I – INFRINGEMENT OF U.S. PATENT NO. 8,510,395

65. The ‘395 Patent is entitled “Contact List Display System And Method.” The application for the ‘395 patent was filed May 5, 2010 and claims priority to provisional and non-provisional filings dated as far back as December 9, 2005. The ‘395 Patent issued on August 13, 2013.

66. Plaintiff is the owner of the ‘395 Patent and it has all substantial rights to the ‘395 Patent, including the right and standing to sue and recover damages for past, present and future infringement of the patent.

67. Claim 1 of the ‘395 Patent covers a system comprising “a network login engine; a network contacts database embodied in one or more non-transitory computer readable mediums; a web server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine: controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider, updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider, maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device.”

68. Claim 7 of the ‘395 Patent covers a method comprising “joining a high level network;

joining a first low level network associated with a first messaging service provider and a second low level network associated with a second messaging service provider; obtaining a first contact list associated with the first messaging service provider; obtaining a second contact list associated with the second messaging service provider; maintaining an aggregated contact [list] that comprises the first contact list and the second contact list; logging into the high level network; displaying the aggregated contact list.”

69. LinkedIn has infringed, and is now infringing, the ‘395 patent, including at least claims 1 and 7, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing, without authority from Plaintiff, systems and methods for obtaining and aggregating contact information from a plurality of messaging services providers via LinkedIn’s LinkedIn Application system, including as claimed in the ‘395 asserted claims. On information and belief, LinkedIn practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising www.linkedin.com.

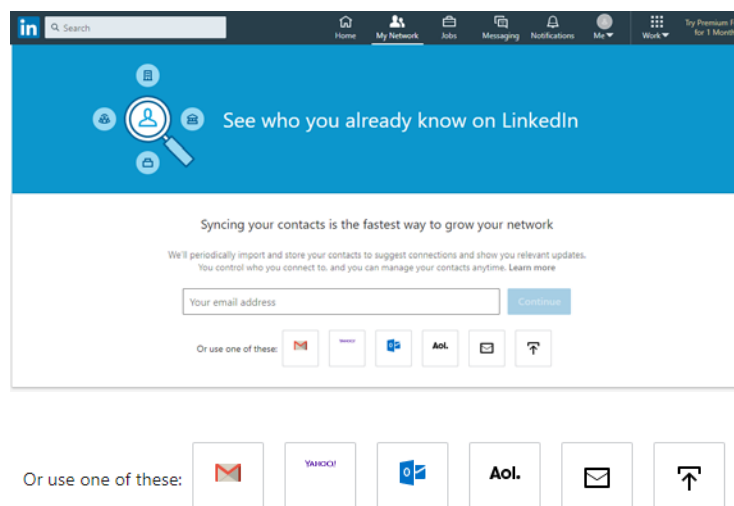
70. Without limitation, the accused system comprising the LinkedIn Application system that comprises a network login engine, a network contacts database, a web server and a contact aggregation engine, wherein the contact aggregation controls the network login engine to controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider, updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider, maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory

computer readable medium at the web server, and provides the aggregated contact list to a display device.

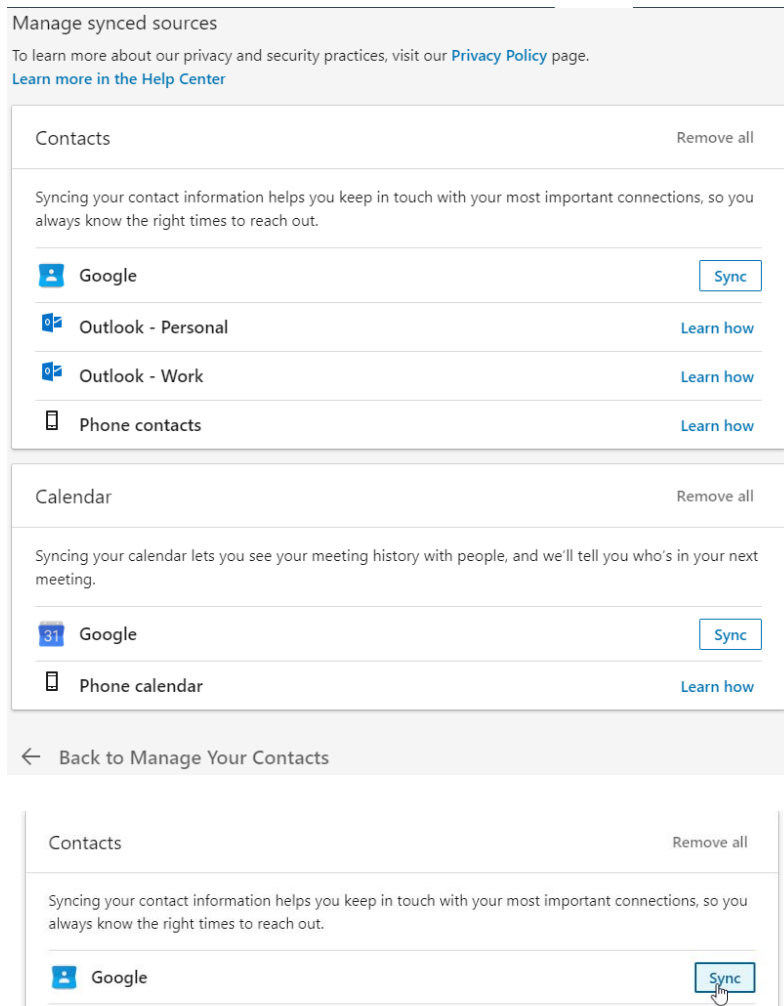
71. Without limitation and for example, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods to permit a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and updating the network contacts.

72. Further, the LinkedIn Application comprises systems and methods which permit a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and update the network contacts.

73. For example, the LinkedIn Application permits a user to join multiple networks associated with respective messaging service providers which interact with, *inter alia*, APIs of LinkedIn and the other networks:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>



See, e.g., LinkedIn User Manage Syncing Settings page at <https://www.linkedin.com/mynetwork/settings/manage-syncing/>

Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.



See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

Syncing Contacts from Other Address Books and Sources

LinkedIn Contacts Manager can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

Note: If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.



To sync contacts:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Add more contacts** on the right rail.

Note: You'll be redirected to a page where you can enter the source you want to import the contacts from.

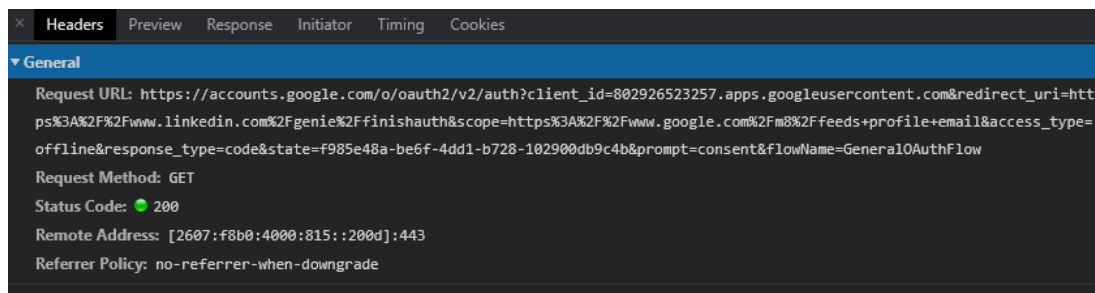
Note: If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

Important: Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>

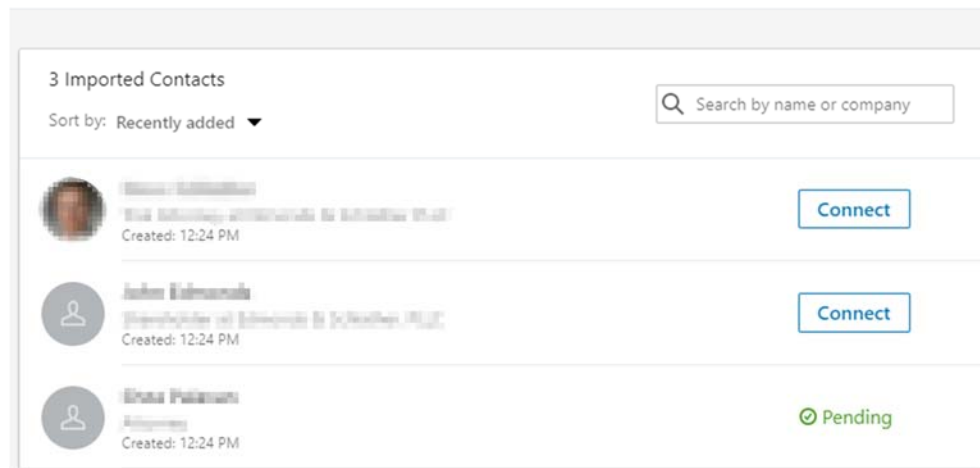


See, e.g., Google Sign-In Request page at https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email

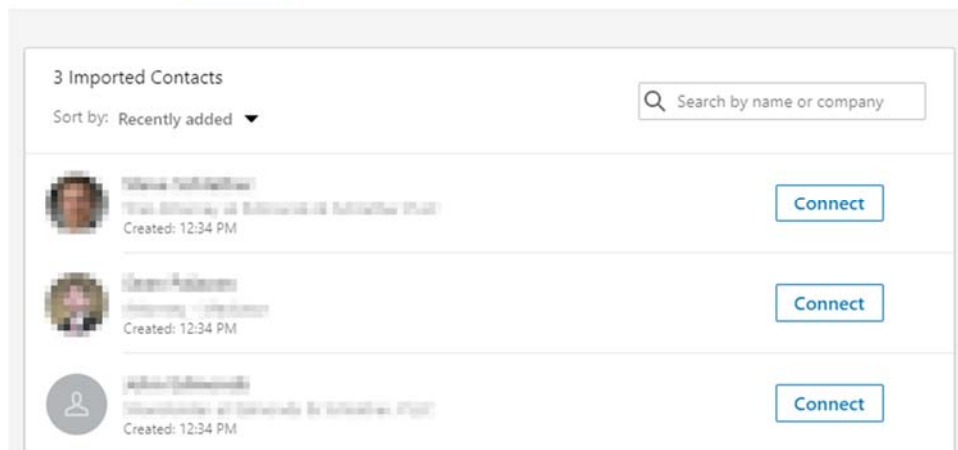
74. For example, the LinkedIn Application obtains a contact list from each of the user's said connected networks for aggregating, maintaining, and displaying:



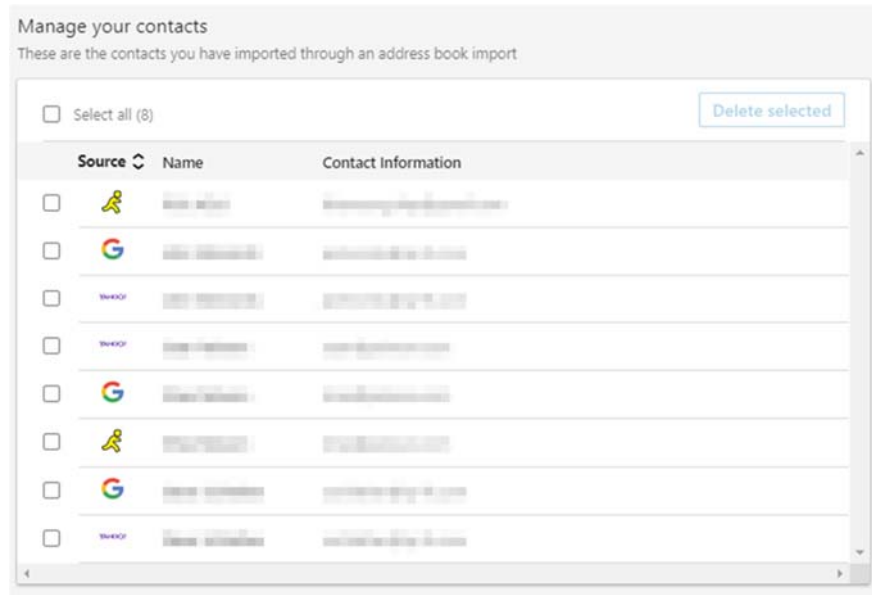
Saved contacts **Google** ▼



Saved contacts **Yahoo** ▼



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



See, e.g., User LinkedIn Manage Your Contacts page at <https://www.linkedin.com/mynetwork/contacts/>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your **contacts and connections** aren't the same thing.

You can begin using LinkedIn Contacts Manager by **importing and inviting your email contacts, creating and uploading a contacts file**, and **syncing contacts from other sources**. Currently, you can only sync Google Calendar and Google Contacts, so **old sources may be missing**.

Once you've synced and imported your contacts, you can **delete specific contacts** to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can **withdraw the invitations**.

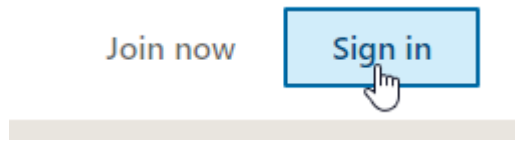
LinkedIn takes your privacy and the security of your data seriously. Learn more about the **privacy of your information in LinkedIn Contacts Manager**, and how you can **access your account data**.

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

75. The LinkedIn Application comprises a network login engine. For example, an engine for

use by the user to log into the user's LinkedIn account and/or the user's social media or other networks or, if the user does not presently have a LinkedIn account, they may create a new account:





Join now Sign in

LinkedIn

Welcome Back

Don't miss your next opportunity. Sign in to stay updated on your professional world.

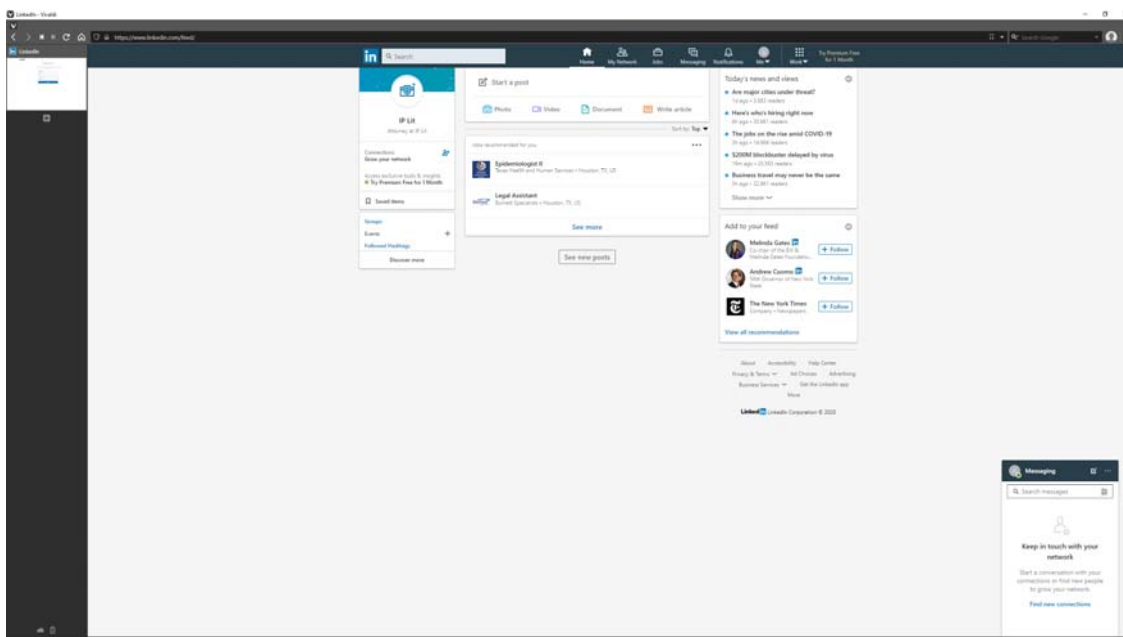
Email or Phone 

Password 

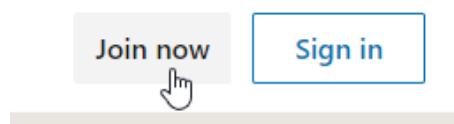
Sign in

[Forgot password?](#)

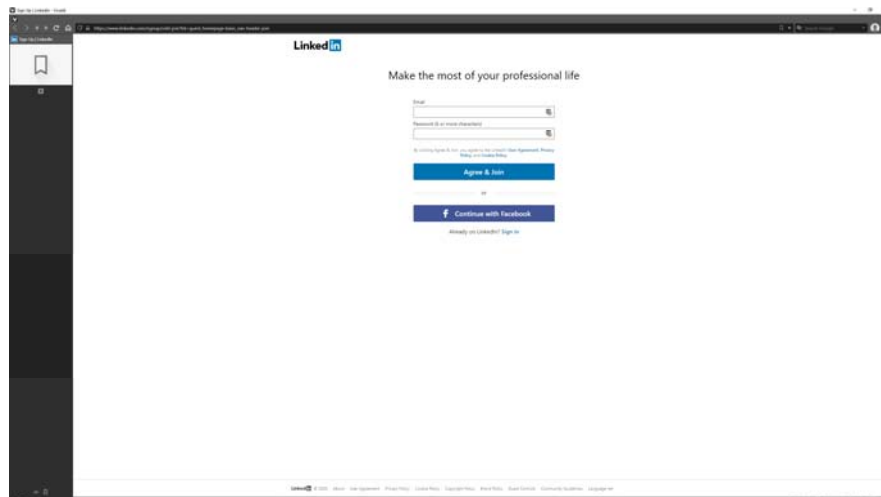
New to LinkedIn? [Join now](#)



See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



Join now Sign in



Welcome, IP!

Let's start your profile, connect to people you know, and engage with them on topics you care about.

Country/Region *

City/District *

Next

Your profile helps you discover the right people and opportunities

Most recent job title *

Most recent company *

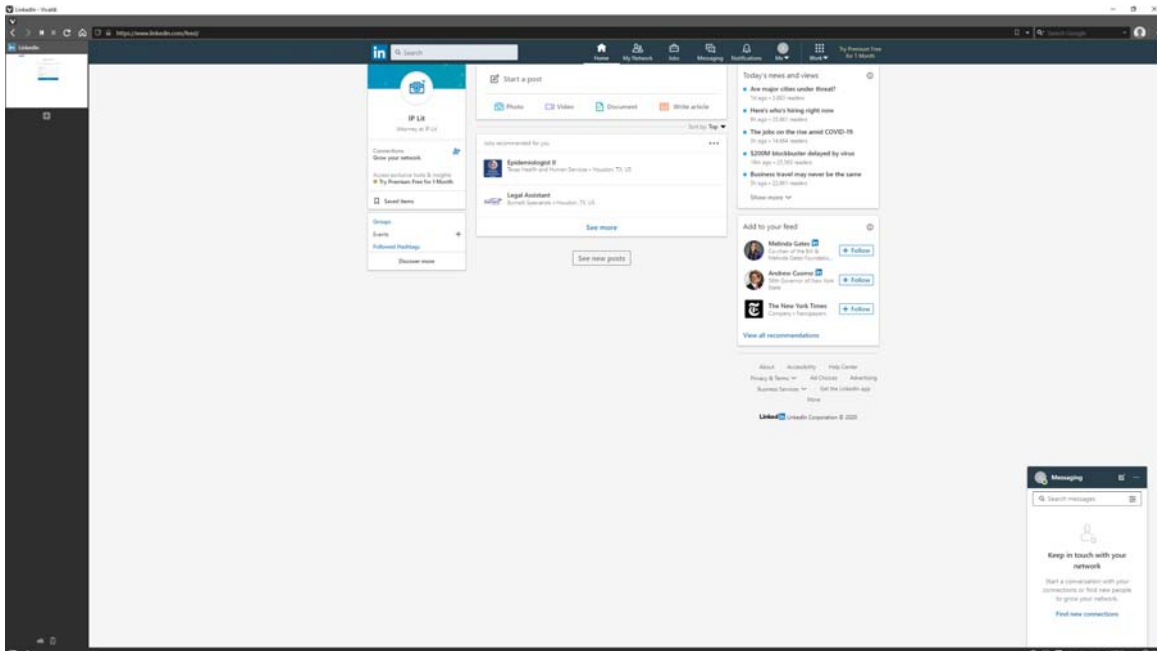
Industry *



There are 3527937 members in the same industry on LinkedIn.

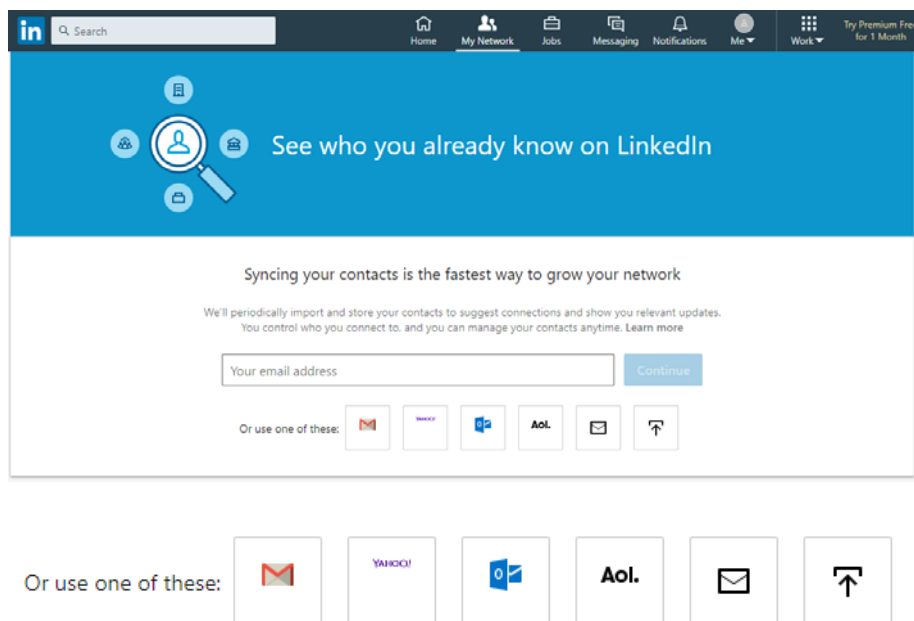
I'm a student

Continue



76. See, e.g., LinkedIn Signup page at <https://www.linkedin.com/signup/cold-join>

77. Further, the LinkedIn Application permits the user to login to a number of different networks, and the user may further connect said network to the user's LinkedIn account, which interacts with APIs of LinkedIn and these other networks and redirects the user to these other networks for signing in:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>

Manage synced sources

To learn more about our privacy and security practices, visit our [Privacy Policy](#) page.

[Learn more in the Help Center](#)

Contacts

Remove all

Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.

Google

Sync

Outlook - Personal

Learn how

Outlook - Work

Learn how

Phone contacts

Learn how

Calendar

Remove all

Syncing your calendar lets you see your meeting history with people, and we'll tell you who's in your next meeting.

Google

Sync

Phone calendar

Learn how

← Back to Manage Your Contacts

Contacts

Remove all

Syncing your contact information helps you keep in touch with your most important connections, so you always know the right times to reach out.

Google

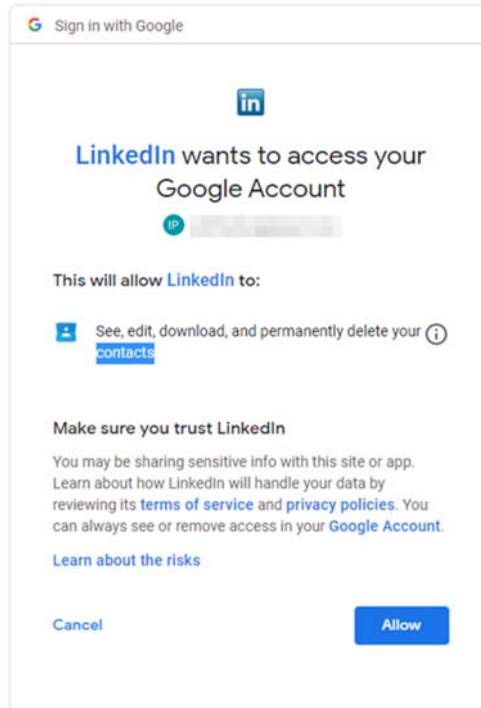
Sync

See, e.g., LinkedIn User Manage Syncing page
<https://www.linkedin.com/mynetwork/settings/manage-syncing/>

```

▼ Query String Parameters  view source  view URL encoded
client_id: 802926523257.apps.googleusercontent.com
redirect_uri: https://www.linkedin.com/genie/finishauth
scope: https://www.google.com/m8/feeds profile email
access_type: offline
response_type: code
state: f985e48a-be6f-4dd1-b728-102900db9c4b
prompt: consent
flowName: GeneralOAuthFlow

```



See, e.g., Google Sign-In Request page at https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email

78. The LinkedIn application comprises a network contacts database embodied in one or more non-transitory computer readable mediums. For example, the LinkedIn Application comprises an address book comprising at least the user's LinkedIn contacts embodied in, for example, a hard drive and/or other ROM and/or, at a minimum, RAM, for storing the user's contact list in said database on LinkedIn's servers:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using **LinkedIn Contacts Manager**. Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your **contacts and connections** aren't the same thing.

You can begin using LinkedIn Contacts Manager by **importing and inviting your email contacts**, **creating and uploading a contacts file**, and **syncing contacts from other sources**. Currently, you can only sync Google Calendar and Google Contacts, so **old sources may be missing**.

Once you've synced and imported your contacts, you can **delete specific contacts** to remove them from your **LinkedIn Contacts Manager address book**. If you've accidentally sent invitations to the contacts you imported, you can **withdraw the invitations**.

LinkedIn takes your privacy and the security of your data seriously. Learn more about the **privacy of your information in LinkedIn Contacts Manager**, and how you can **access your account data**.

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our **Privacy Policy**.

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.



See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

Syncing Contacts from Other Address Books and Sources

LinkedIn Contacts Manager can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

Note: If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.



To sync contacts:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Add more contacts** on the right rail.

Note: You'll be redirected to a page where you can enter the source you want to import the contacts from.

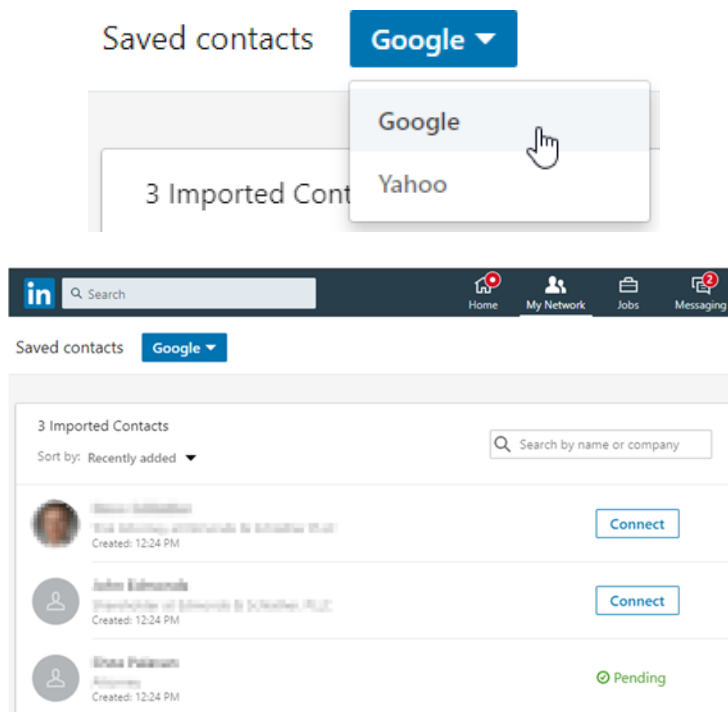
Note: If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

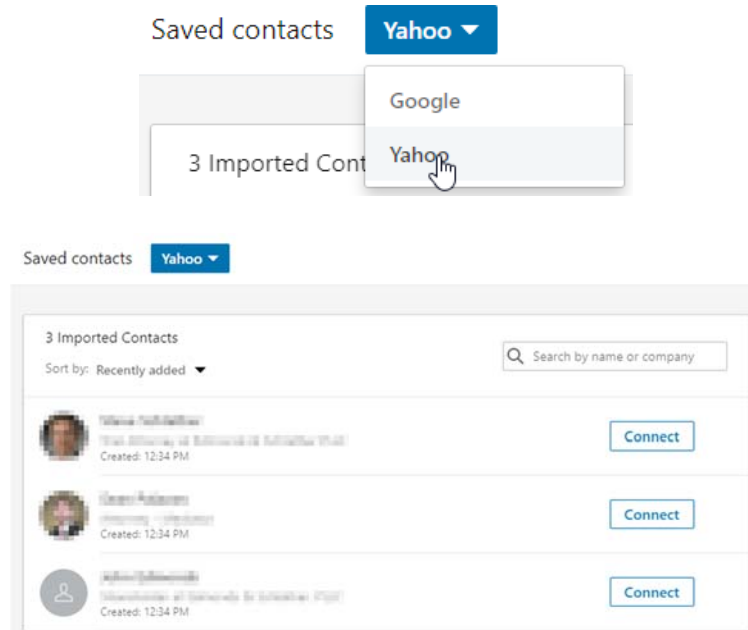
Important: Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

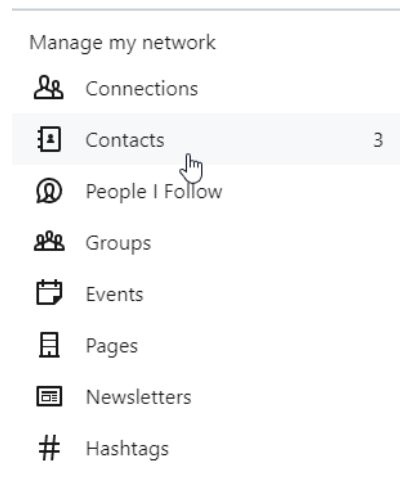
Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>



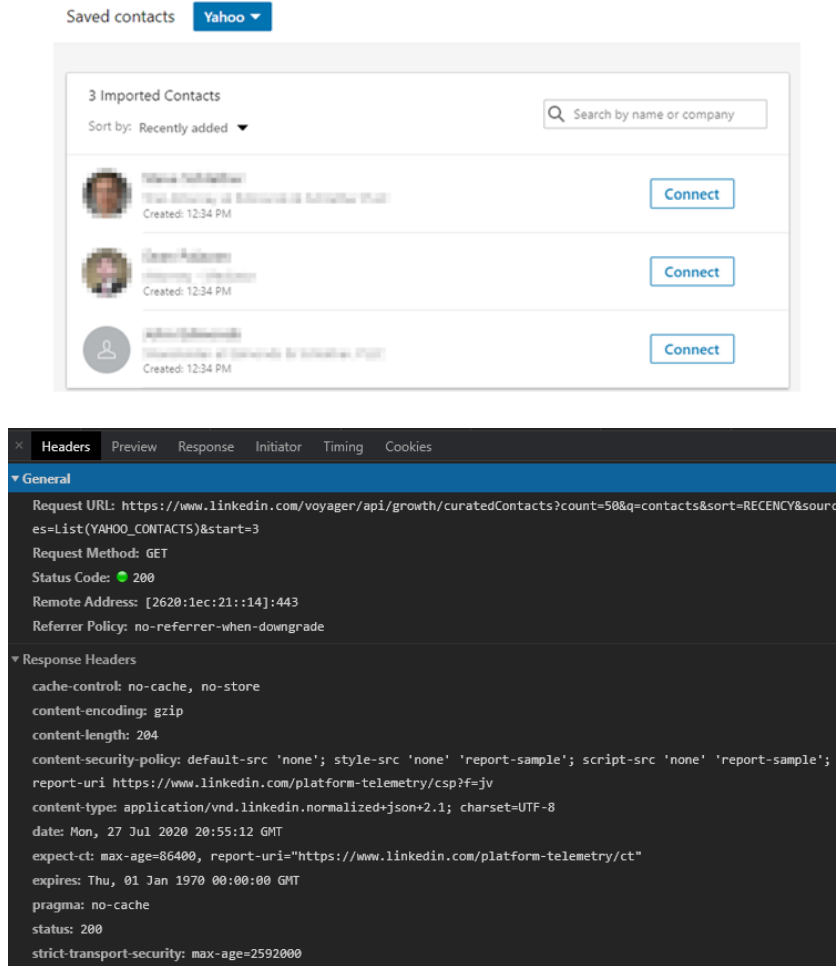


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



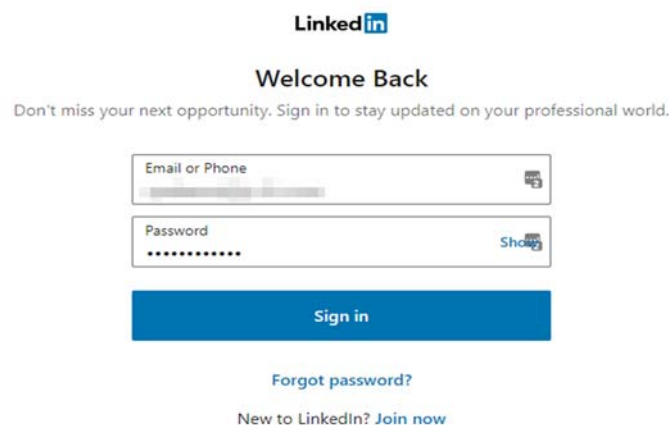
See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>

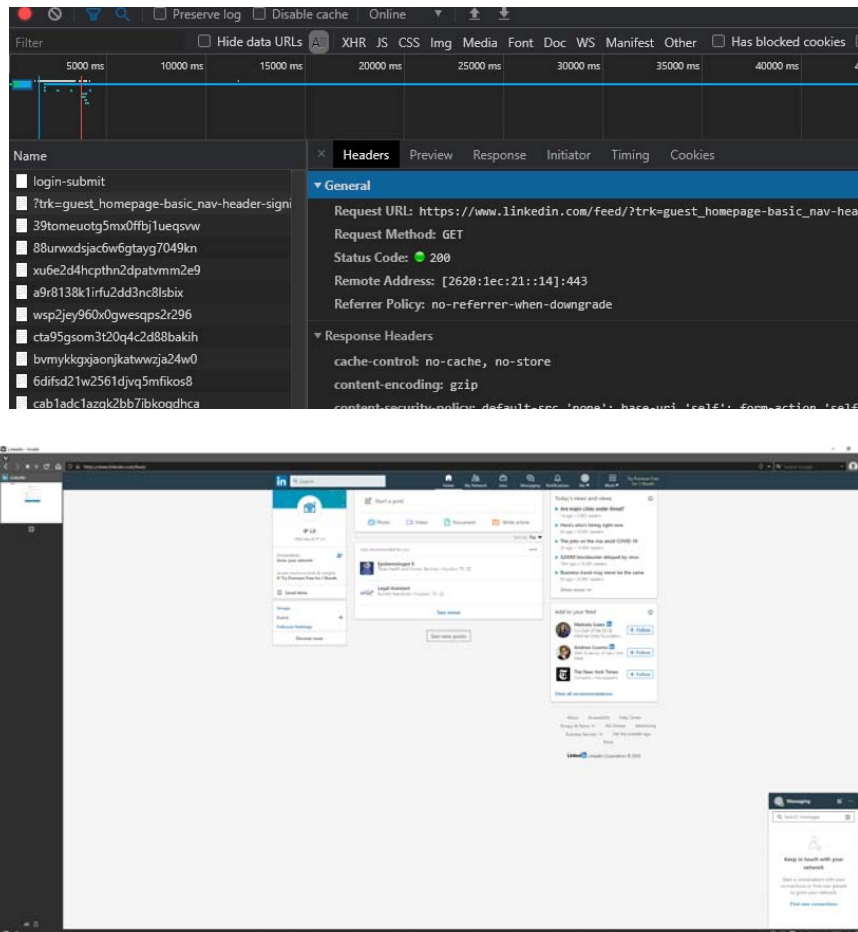




See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

79. The LinkedIn Application comprises a web server coupled to the network contacts database. For example, the LinkedIn Application comprises a web server connected to the LinkedIn contact database:





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>

80. The LinkedIn Application comprises a contact aggregation engine coupled to the network login engine and the network contacts database. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager which permits LinkedIn's servers to aggregate the user's contacts and is coupled to the LinkedIn login engine and to the storage medium comprising the user's LinkedIn contacts:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using **LinkedIn Contacts Manager**. Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your **contacts and connections** aren't the same thing.

You can begin using LinkedIn Contacts Manager by **importing and inviting your email contacts**, **creating and uploading a contacts file**, and **syncing contacts from other sources**. Currently, you can only sync Google Calendar and Google Contacts, so **old sources may be missing**.

Once you've synced and imported your contacts, you can **delete specific contacts** to remove them from your **LinkedIn Contacts Manager address book**. If you've accidentally sent invitations to the contacts you imported, you can **withdraw the invitations**.

LinkedIn takes your privacy and the security of your data seriously. Learn more about the **privacy of your information in LinkedIn Contacts Manager**, and how you can **access your account data**.

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our **Privacy Policy**.

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.



See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

Syncing Contacts from Other Address Books and Sources

LinkedIn Contacts Manager can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

Note: If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.



To sync contacts:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the  **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click  **Add more contacts** on the right rail.

Note: You'll be redirected to a page where you can enter the source you want to import the contacts from.

Note: If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

Important: Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>

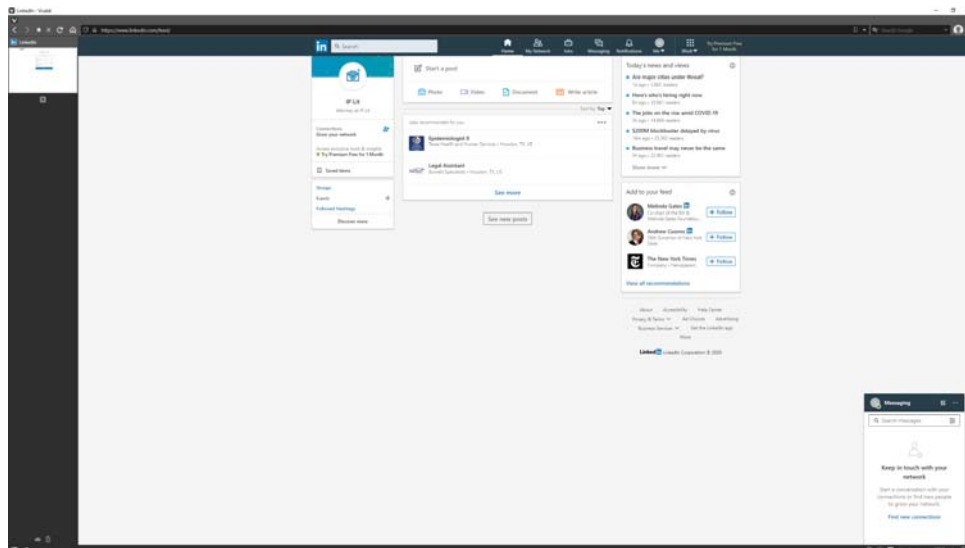
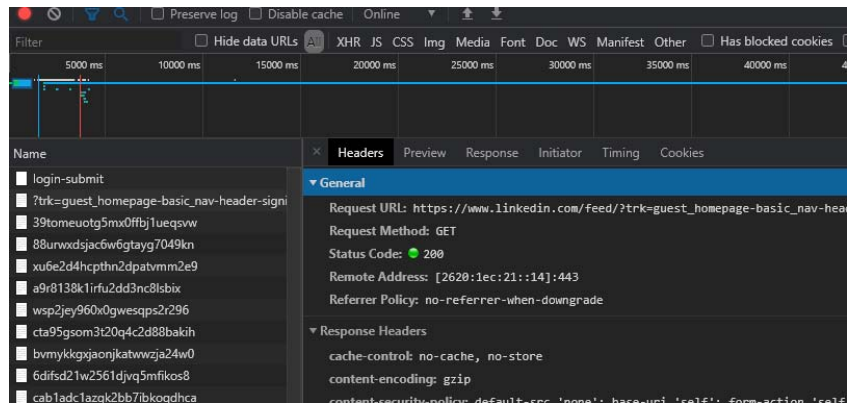


Welcome Back

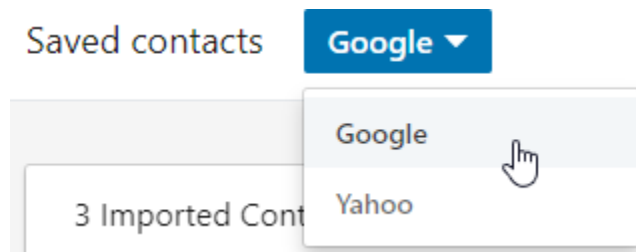
Don't miss your next opportunity. Sign in to stay updated on your professional world.

[Forgot password?](#)

New to LinkedIn? [Join now](#)

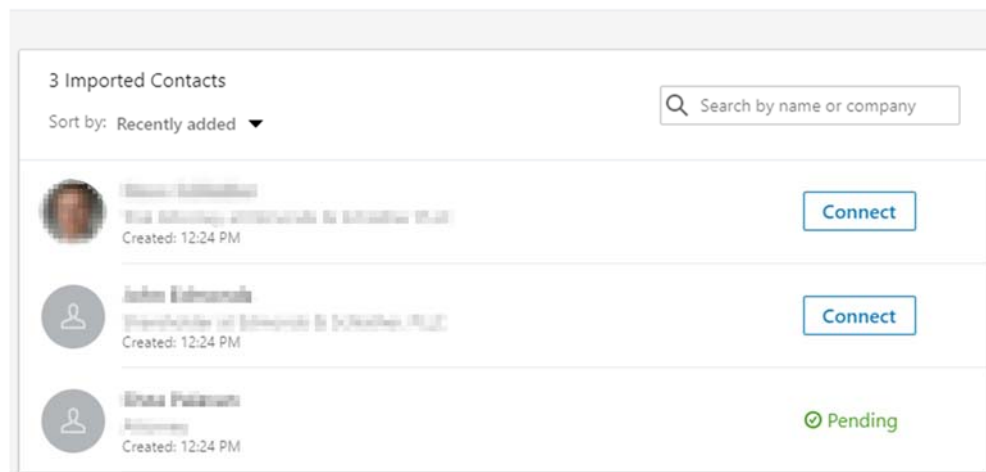


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>





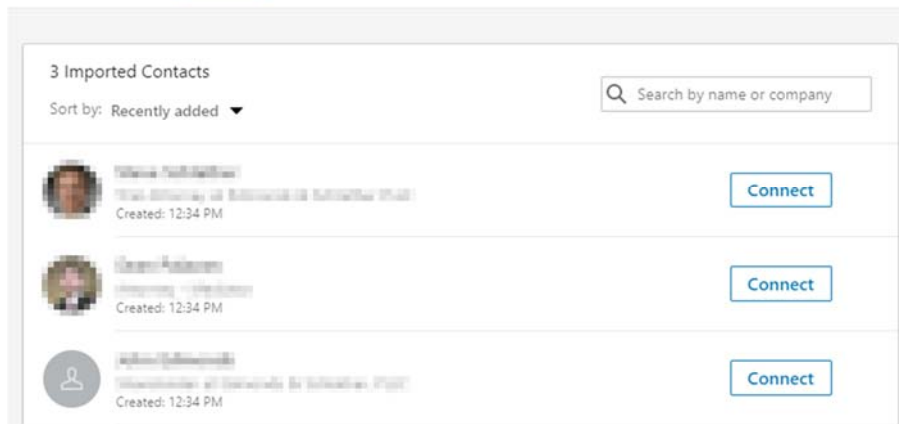
Saved contacts **Google** ▼

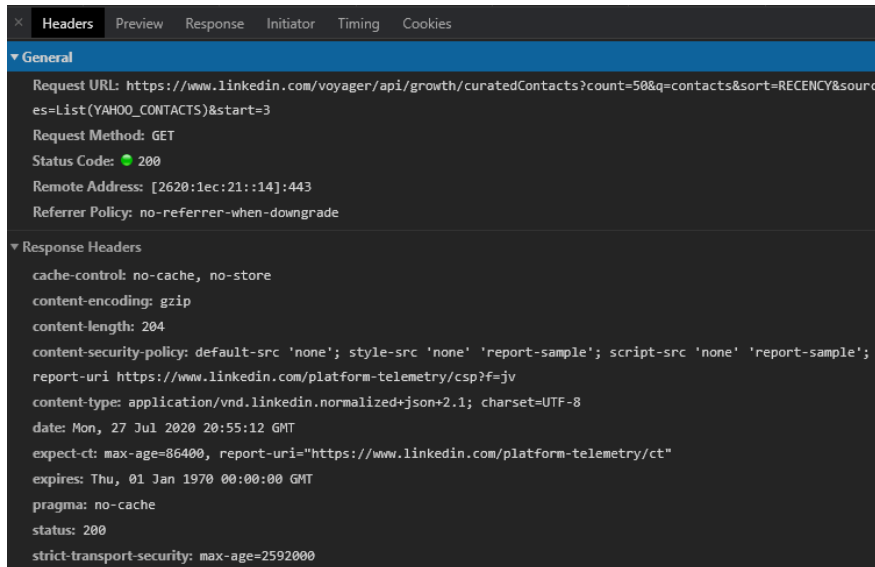


Saved contacts **Yahoo** ▼



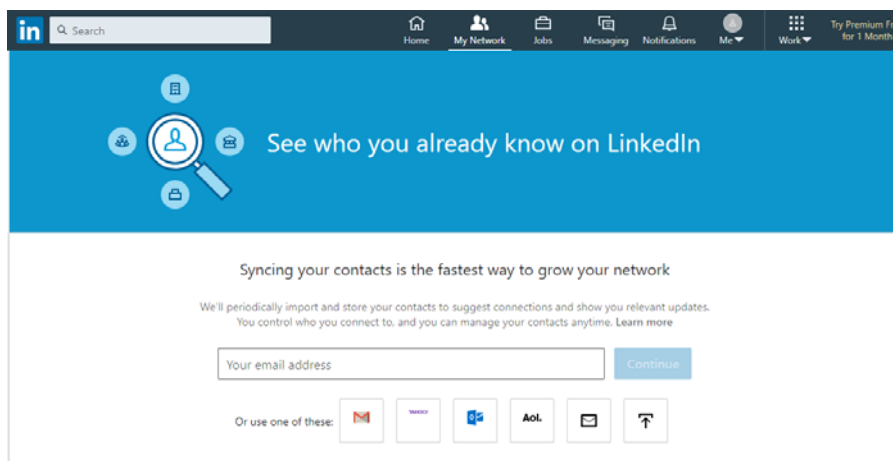
Saved contacts **Yahoo** ▼





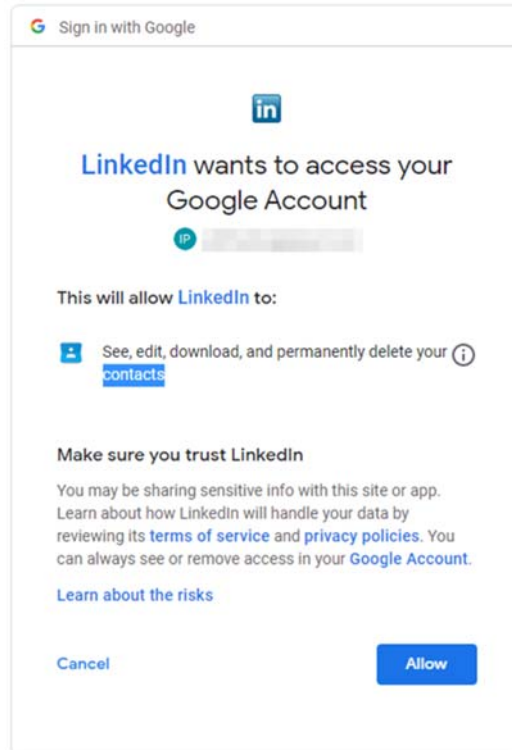
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

81. In operation, the contact aggregation engine (*see* above) controls the network login engine to login or facilitate login to a first network associated with a first messaging service provider and a second network associated with a second messaging service provider. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including facilitating the user logging into and associating the user's various social media and/or other networks, including via APIs of LinkedIn and/or APIs of the user's social media and/or other networks:

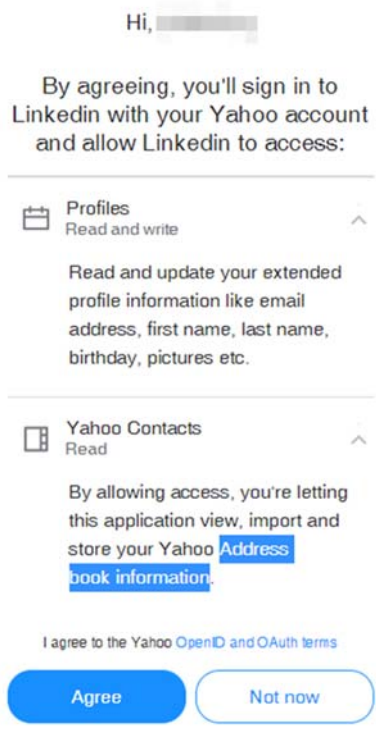




See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>



See, e.g., Google Sign-In Request page at https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email



See, e.g., Yahoo! Sign-In Request page at

https://api.login.yahoo.com/oauth2/request_auth_fe?client_id=dj0yJmk9S0c3Y2RXVmw1RTQxJmQ9WVdrOVIYWmxXbWxDtXpBbWNHbzlnQS0tJnM9Y29uc3VtZXJzZWNYZXQmeD02MA--&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response_type=code&state=5ef8e091-7714-423c-a57b-f607710145fb&nonce=6612439201265914688&guccounter=1&guce_referrer=aHR0cHM6Ly9sb2dpbi55YWhvby5jb20v&guce_referrer_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoLiDu6onlExDhf7tsoQ-EJKVkkqOu2ac4W3pj3P34o_G23xo8DLX4pMAqIKiP6olGZjJKhodFpcMc2yR

82. In operation, the contact aggregation engine (*see* above) updates the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including updating the networks contacts database with contact information obtained from the first messaging service provider and the second messaging service provider,

including obtaining and importing the contacts of the first network onto LinkedIn's servers after the user logs in to the first network and the contacts of the second network onto LinkedIn's servers after the user logs in to the second:

Syncing Contacts from Other Address Books and Sources

LinkedIn Contacts Manager can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the [privacy of the information you sync](#).

Note: If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the **My Network** icon at the top of your LinkedIn homepage.
 2. Click **Contacts** under **Manage my network** on the left rail.
 3. Click **Add more contacts** on the right rail.
- Note:** You'll be redirected to a page where you can enter the source you want to import the contacts from.

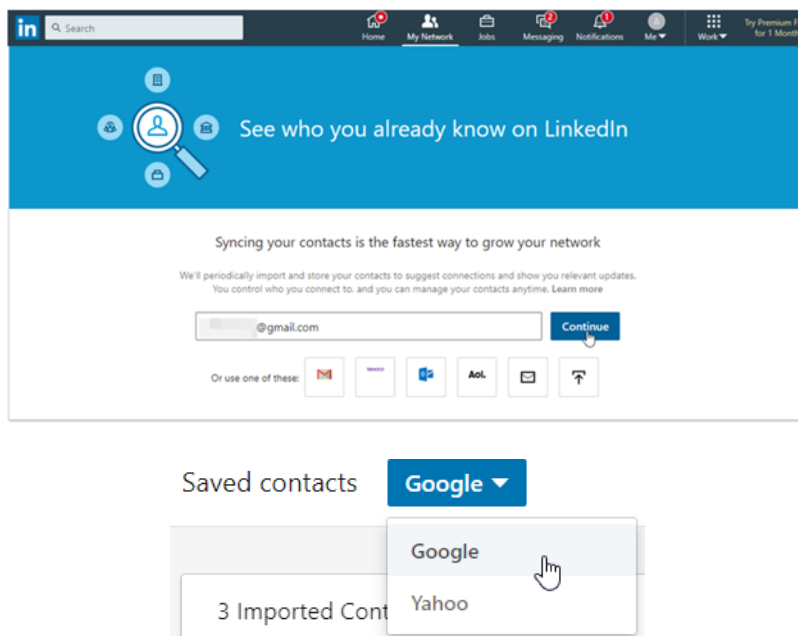
Note: If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

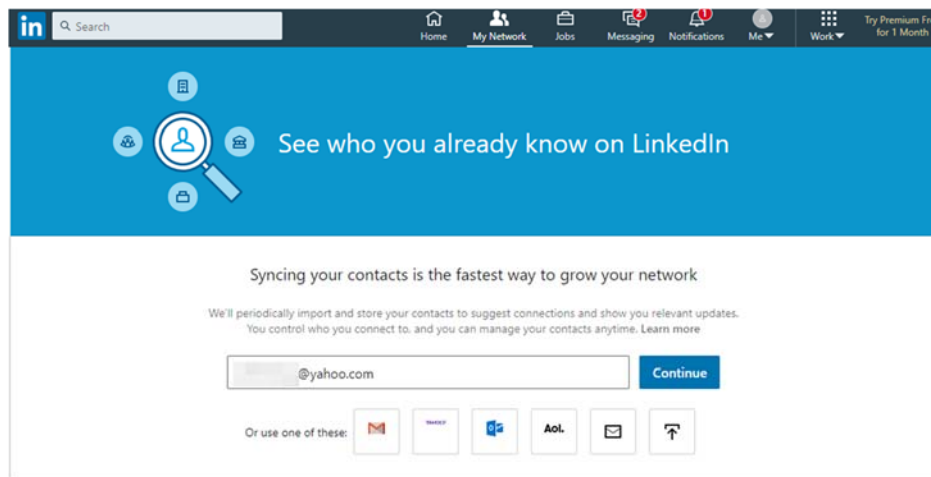
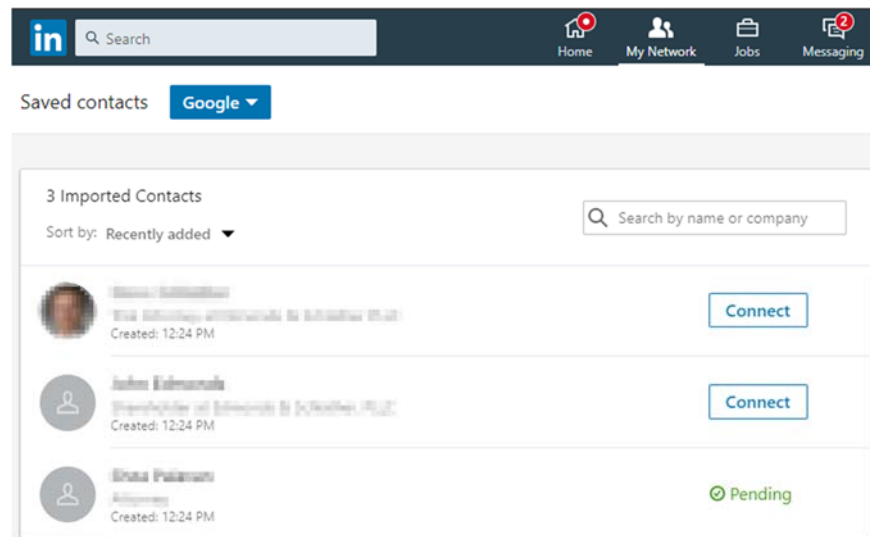
Important: Synced contacts are not automatically invited to connect with you on LinkedIn.

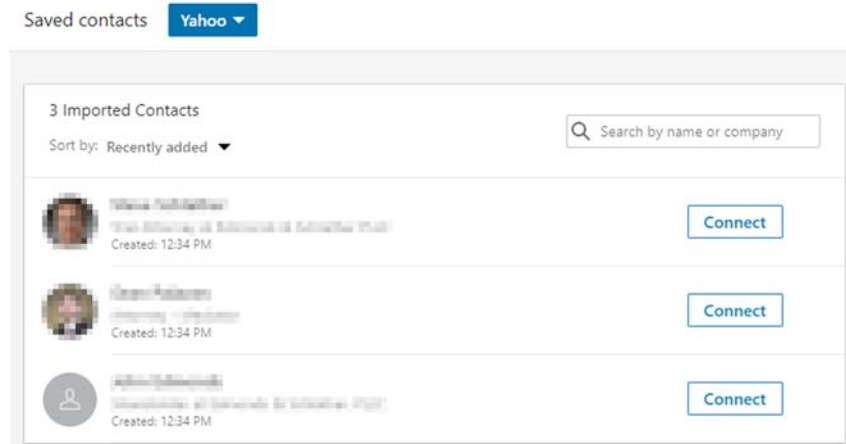
Learn more about [deleting imported contacts](#), and why [old sources may be missing from the Contacts Syncing page](#).

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>







See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

83. In operation, the contact aggregation engine (*see* above) maintains an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including maintaining an aggregated contact list that comprises a first contact list associated with the contact information from the first messaging service provider and a second contact list associated with the contact information from the second messaging service provider, stores the aggregated contact list in a non-transitory computer readable medium at the web server, and provides the aggregated contact list to a display device:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using **LinkedIn Contacts Manager**. Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your **contacts and connections** aren't the same thing.

You can begin using LinkedIn Contacts Manager by **importing and inviting your email contacts**, **creating and uploading a contacts file**, and **syncing contacts from other sources**. Currently, you can only sync Google Calendar and Google Contacts, so **old sources may be missing**.

Once you've synced and imported your contacts, you can **delete specific contacts** to remove them from your **LinkedIn Contacts Manager address book**. If you've accidentally sent invitations to the contacts you imported, you can **withdraw the invitations**.

LinkedIn takes your privacy and the security of your data seriously. Learn more about the **privacy of your information in LinkedIn Contacts Manager**, and how you can **access your account data**.

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our **Privacy Policy**.

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

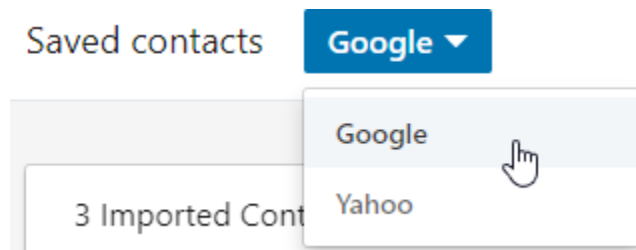
See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

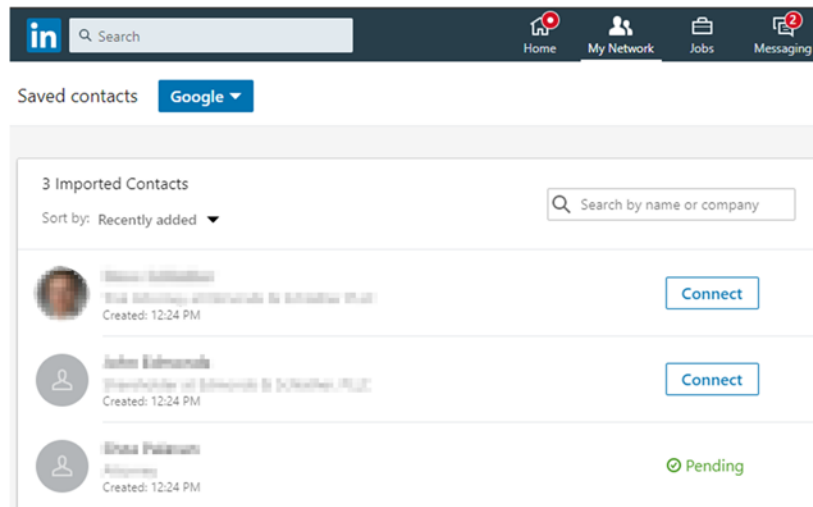
Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

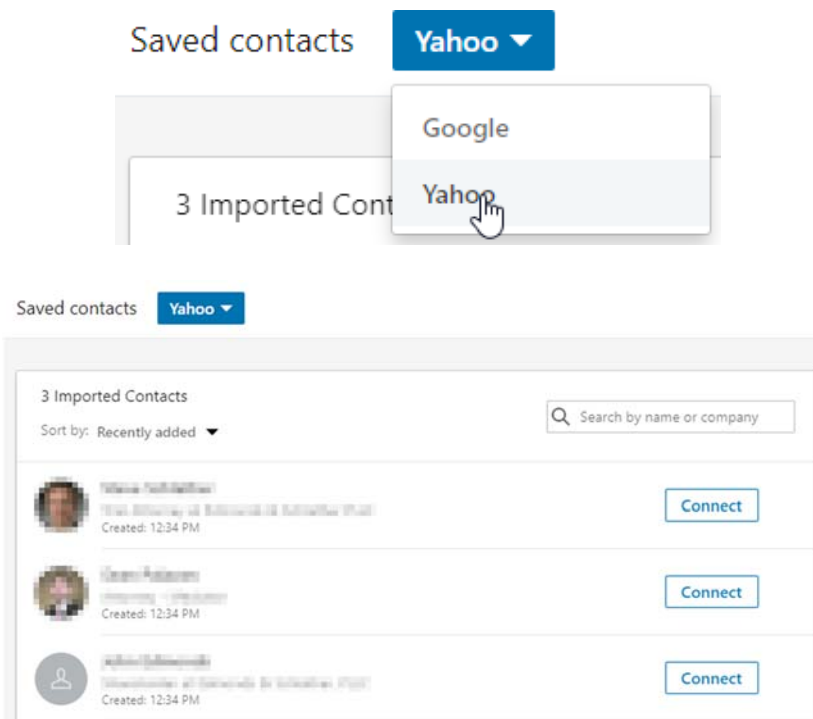
Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>





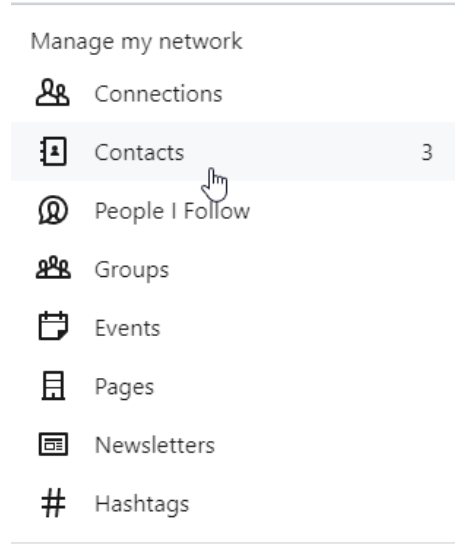
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



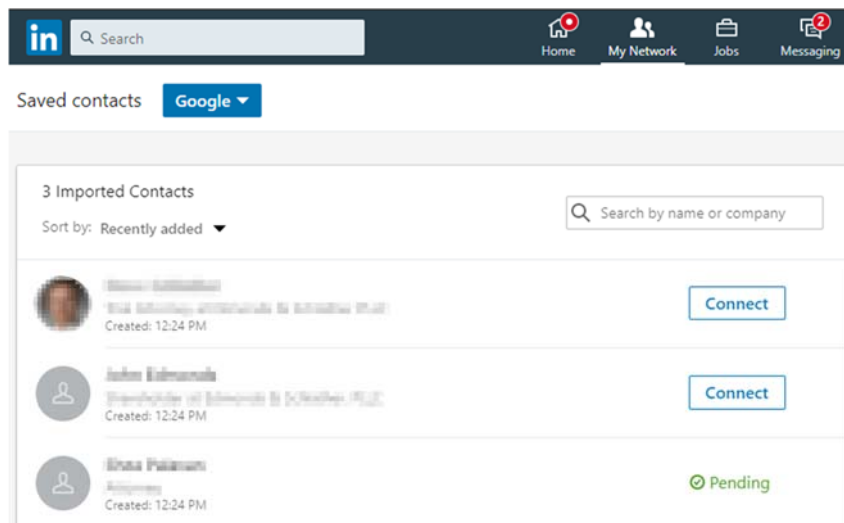
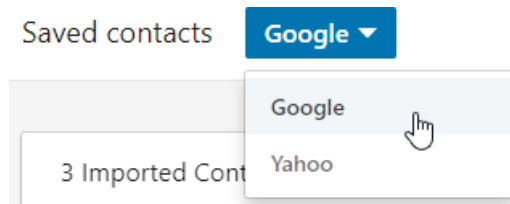
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

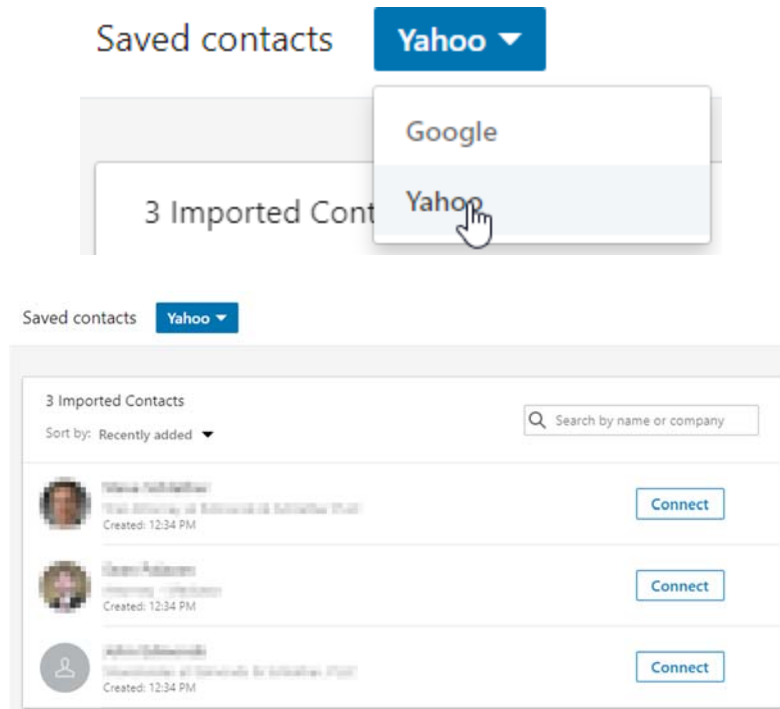
84. In operation, the contact aggregation engine (*see above*) stores the aggregated contact list in a non-transitory computer readable medium at the web server. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including storing the user's aggregated contact list on

LinkedIn's servers:

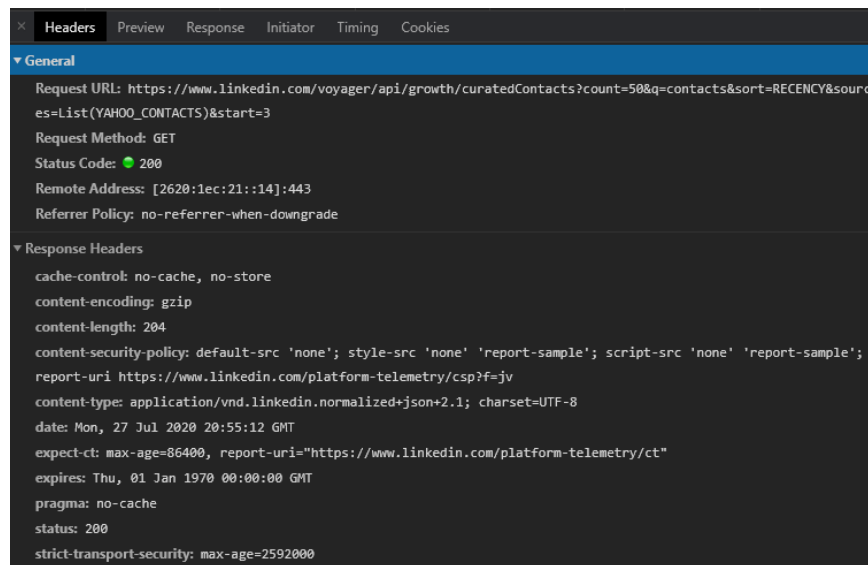


See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>





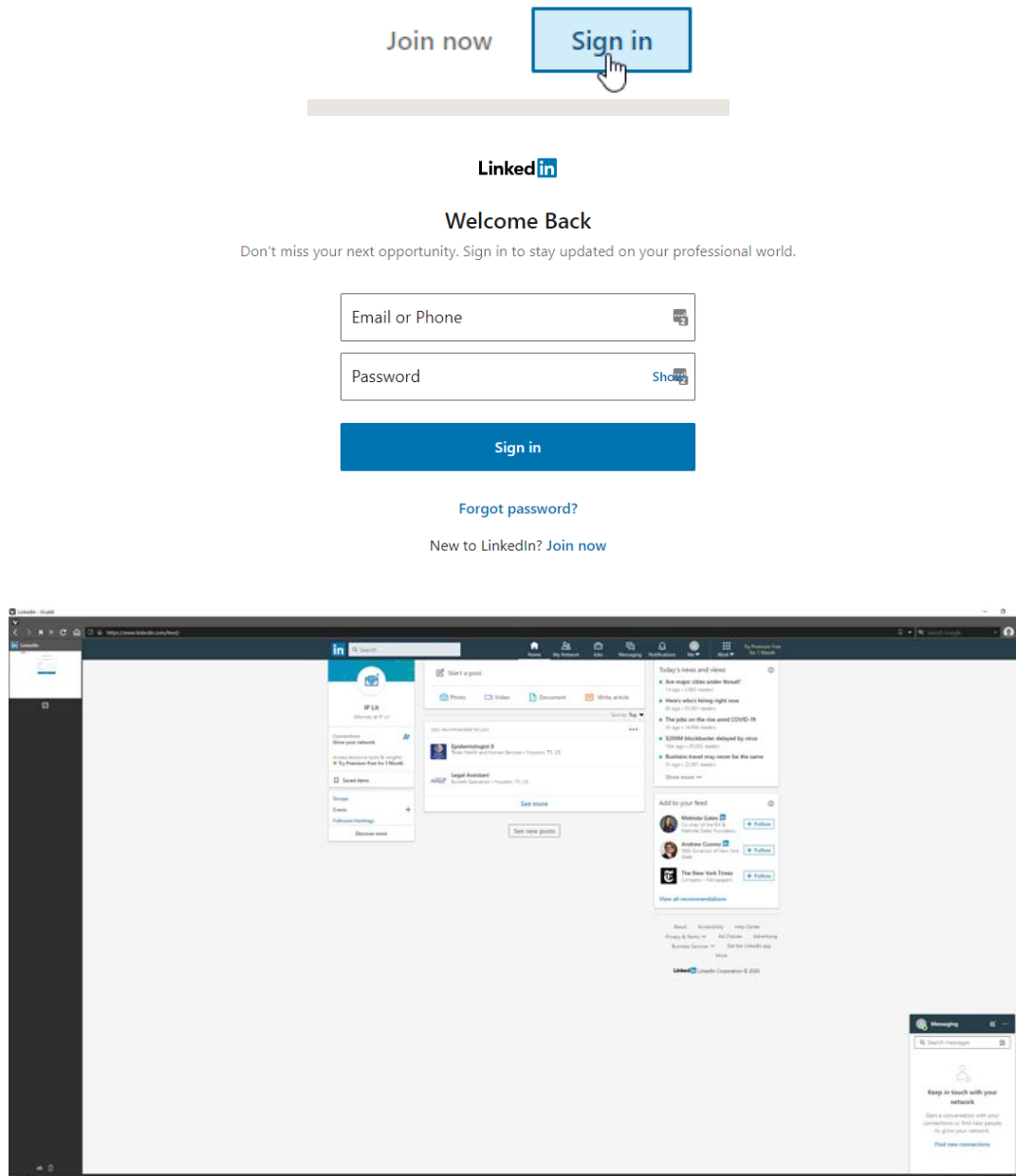
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



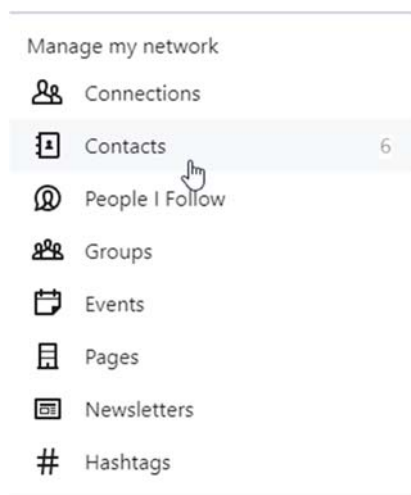
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

85. In operation, the contact aggregation engine (*see* above) provides the aggregated contact list to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including

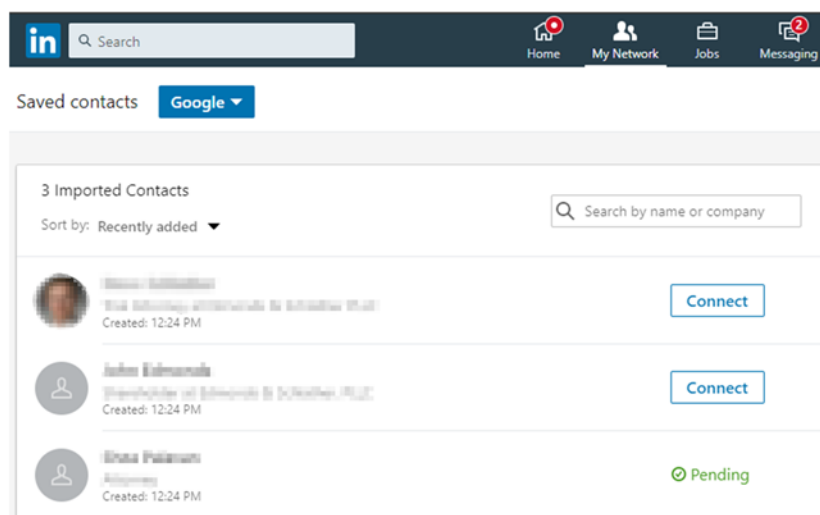
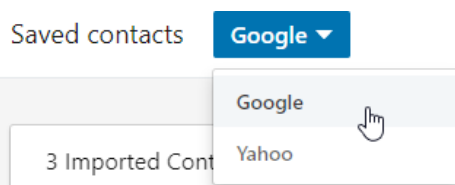
providing the user's aggregated contact list stored on LinkedIn's servers to the user's device for display to the user:

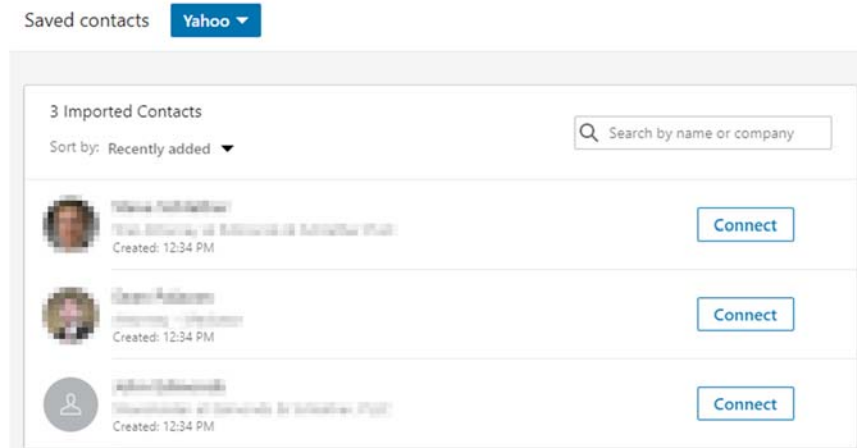


See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

86. LinkedIn has directly infringed, and continues to directly infringe, the claims of the ‘395 Patent, including at least those noted above, including by making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a). Further, including at least to the extent LinkedIn provides and/or supplies software running on a user’s computer, the direct infringement of users that occurs in connection with LinkedIn’s applications and/or web services occurs under the direction or control of LinkedIn.

87. Additionally, or in the alternative, since receiving notice of the ‘395 patent, including if necessary from this suit, LinkedIn has induced, and continues to induce, infringement of the ‘395 Patent in this judicial district, and elsewhere, by actively inducing direct infringement of the ‘395 Patent, including by knowingly and actively aiding or abetting infringement by users, by and through at least instructing and encouraging the use of the LinkedIn products and software noted herein, including the LinkedIn Application system. Such aiding and abetting comprises providing software, web servers, and/or instructions regarding the use and/or operation of the LinkedIn Application system, applications, and web servers in an infringing manner. Such induced infringement has occurred since LinkedIn became aware of the ‘395 Patent, at a minimum, as noted above, and the knowledge and awareness that such actions by users comprise infringement

of the '395 Patent.

88. LinkedIn has had at least constructive notice of the '395 Patent since at least its issuance. LinkedIn will have been on actual notice of the '395 Patent since, at the latest, the service of this Complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the '395 Patent.

89. The LinkedIn Application system clearly meets the asserted claim limitations in their normal and expected usage. On information and belief, normal and expected usage of the LinkedIn Application system by customers and/or end users satisfies the claim limitations for direct infringement. Further, at minimum, the provision of products, systems and/or functionalities clearly capable of such infringing usage and/or provision of instructions/specifications for such infringing usage constitutes inducement of directly infringing usage.

90. Further, as noted above, LinkedIn is being made aware of infringement of the '395 Patent through use of the LinkedIn Application system at least via the infringement allegations set forth herein. Such direct and induced infringement has been and remains clear, unmistakable and inexcusable. On information and belief, LinkedIn knew or should have known of the clear, unmistakable and inexcusable direct and induced infringing conduct at least receiving notice of the '395 Patent. Thus, on information and belief, Defendants have, since receiving notice of the '395 Patent, specifically intended to induce direct infringement by customers and/or end users.

91. EBT believes and contends that, at a minimum, LinkedIn's knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the '395 Patent since receiving notice of its infringement of the '395 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a

finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the ‘395 Patent.

COUNT II – INFRINGEMENT OF U.S. PATENT NO. 9,584,453

92. The ‘453 Patent is entitled “Contact List Display System And Method.” The application for the ‘453 patent was filed May 5, 2010 and claims priority to provisional and non-provisional filings dated as far back as December 9, 2005. The ‘453 Patent issued on August 13, 2013.

93. Plaintiff is the owner of the ‘453 Patent and it has all substantial rights to the ‘453 Patent, including the right and standing to sue and recover damages for past, present and future infringement of the patent.

94. Claim 1 of the ‘453 Patent covers a system comprising “a network interface; a network login engine coupled to the network interface; a network contacts database embodied in one or more non-transitory computer-readable mediums; a server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.”

95. LinkedIn has infringed, and is now infringing, the ‘453 patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising

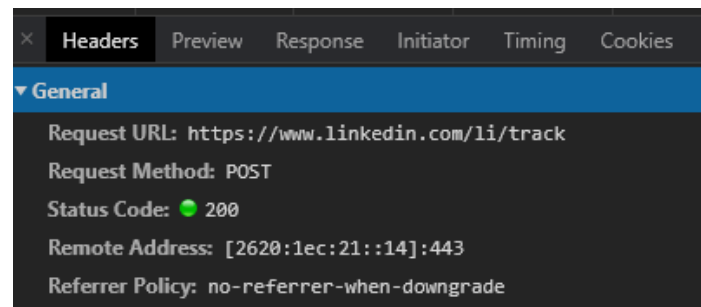
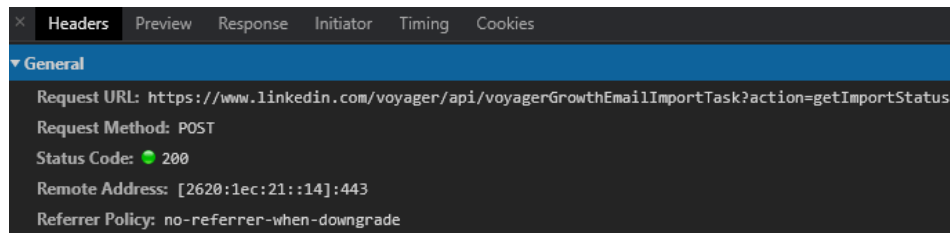
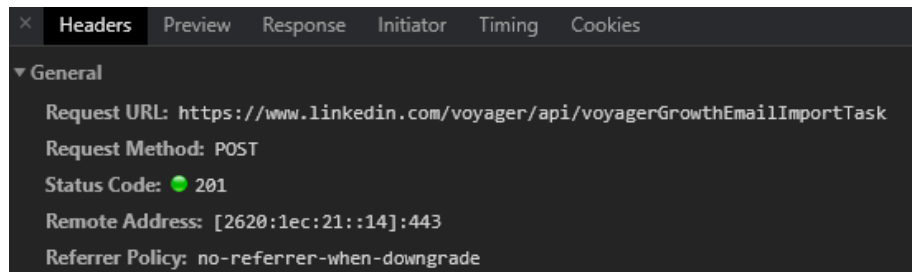
the practicing, without authority from Plaintiff, systems and methods for obtaining and aggregating contact information from a plurality of messaging services providers via LinkedIn's LinkedIn Application system, including as claimed in the '453 asserted claims. On information and belief, LinkedIn practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising www.linkedin.com.

96. Without limitation, the accused system comprising the LinkedIn Application system comprises a network interface; a network login engine coupled to the network interface; a network contacts database embodied in one or more non-transitory computer-readable mediums; a server coupled to the network contacts database; a contact aggregation engine coupled to the network login engine and the network contacts database; wherein, in operation, the contact aggregation engine controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging services through a high level network using the network interface to access contact information from the plurality of messaging services, updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list, stores the aggregated contact list in a non-transitory computer-readable medium at the server, and provides the aggregated contact list including the contact information to a display device.

97. For example, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods permits a user to login via the LinkedIn Application to a first and second messaging service provider, obtain a first contact list from the first messaging service provider and a second contact list from the second messaging service provider, and update the network contacts.

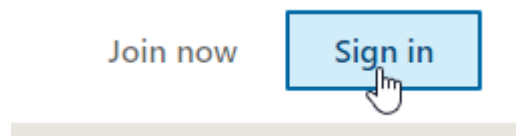
98. The LinkedIn Application comprises a network interface. For example, the LinkedIn

Application comprises ethernet, fiber, and/or other network connectivity for facilitating LAN, WAN, and/or other network activity:



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/>

99. The LinkedIn Application comprises a network login engine coupled to the network interface. For example, the LinkedIn Application comprises the LinkedIn login webservice comprising an engine for use by the user to log into the user's LinkedIn account and/or the user's social media or other networks coupled to the LinkedIn servers:





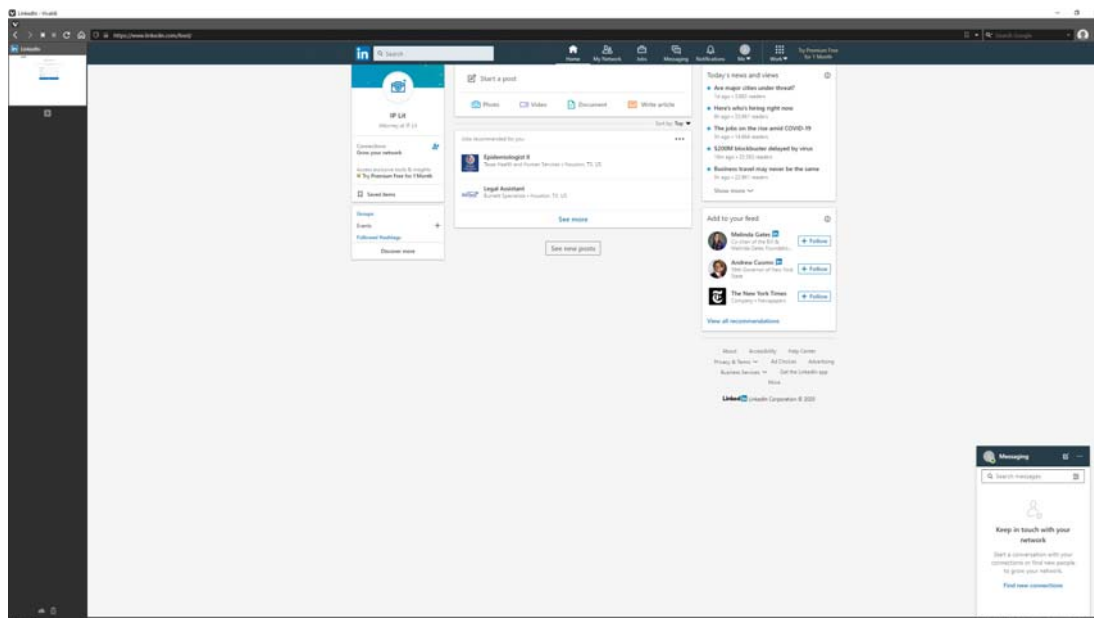
Welcome Back

Don't miss your next opportunity. Sign in to stay updated on your professional world.

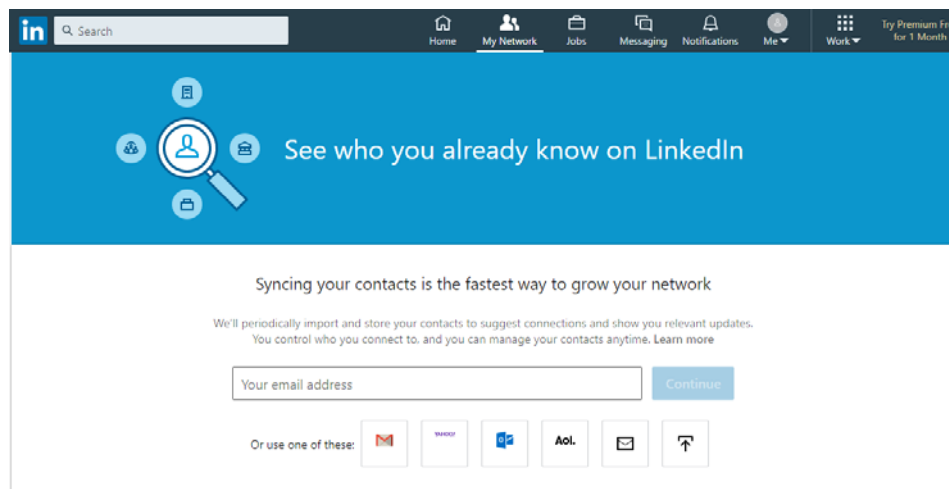
[Show](#)

[Forgot password?](#)

New to LinkedIn? [Join now](#)



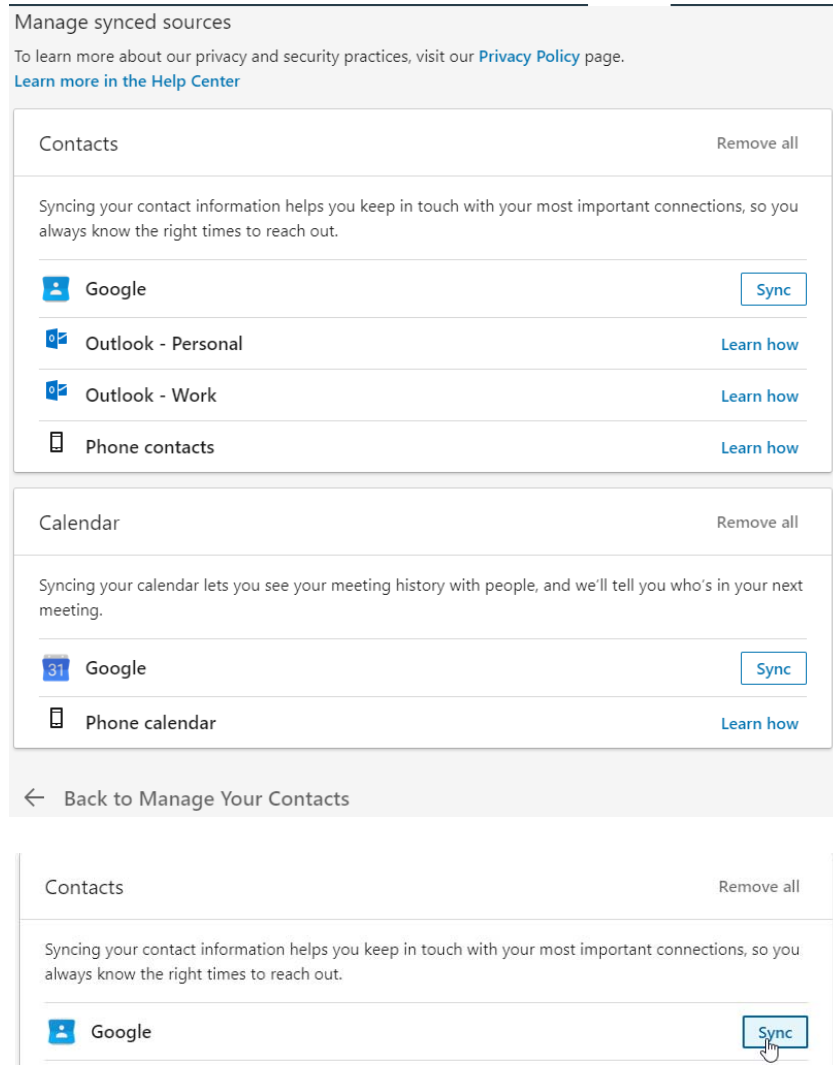
See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



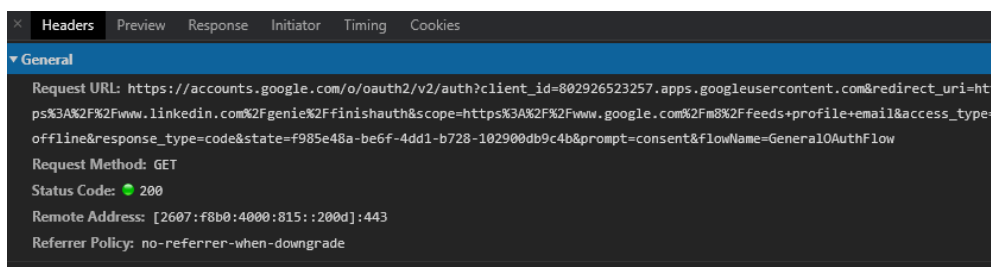
Or use one of these:

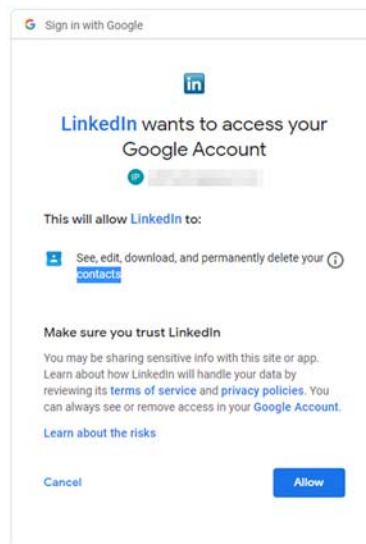


See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>

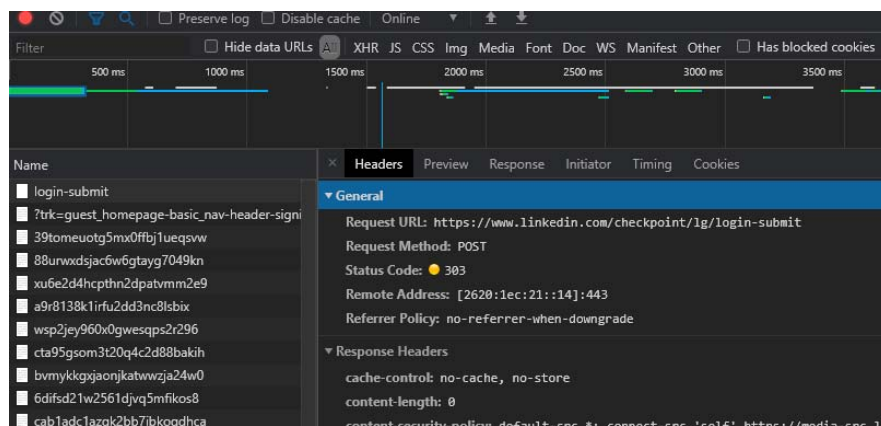


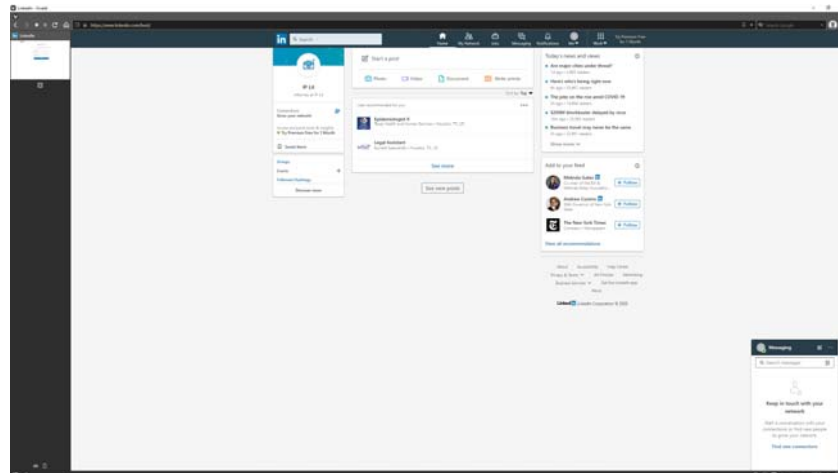
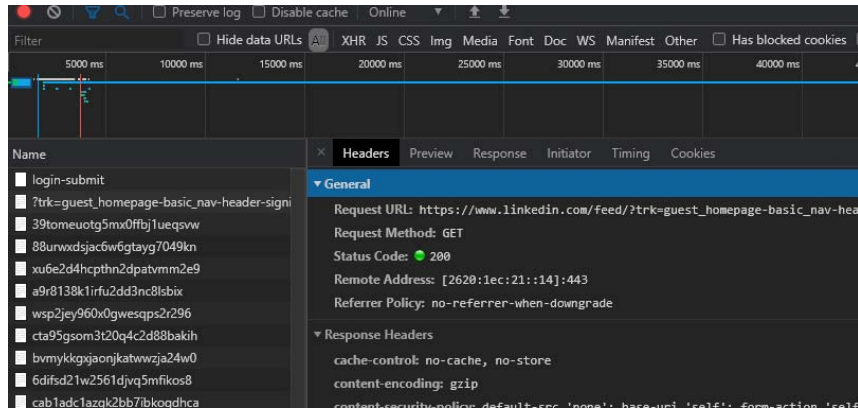
See, e.g., LinkedIn User Manage Syncing page
<https://www.linkedin.com/mynetwork/settings/manage-syncing/>



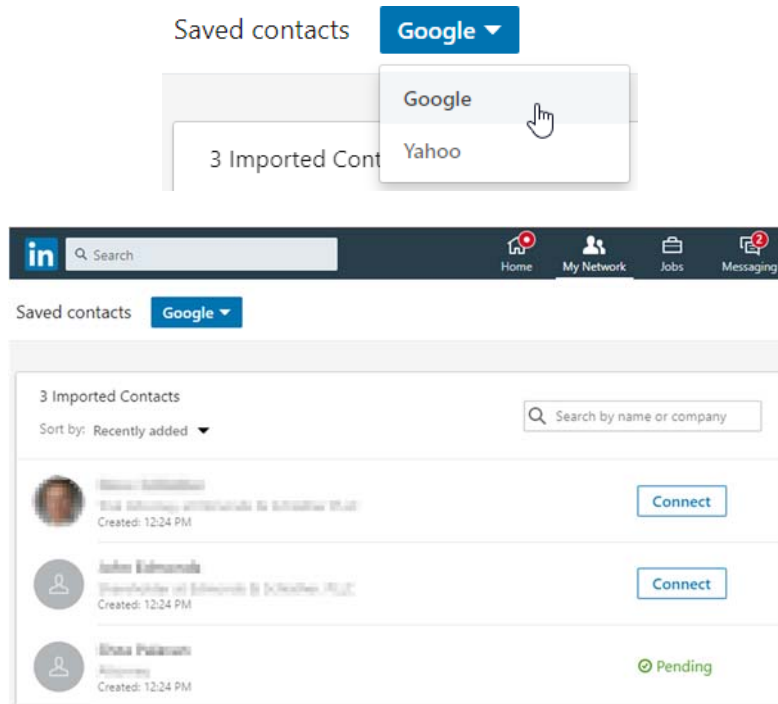


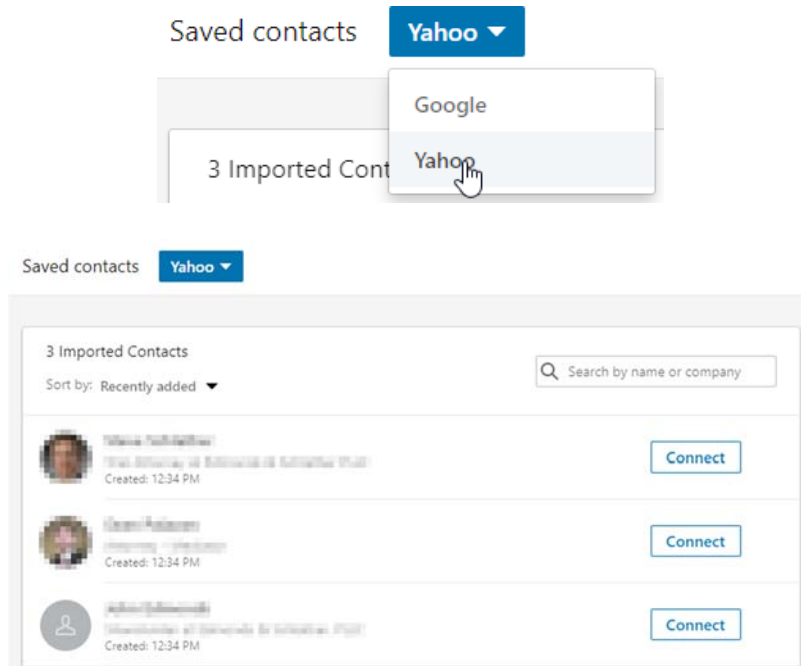
See, e.g., Google Sign-In Request page at https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email



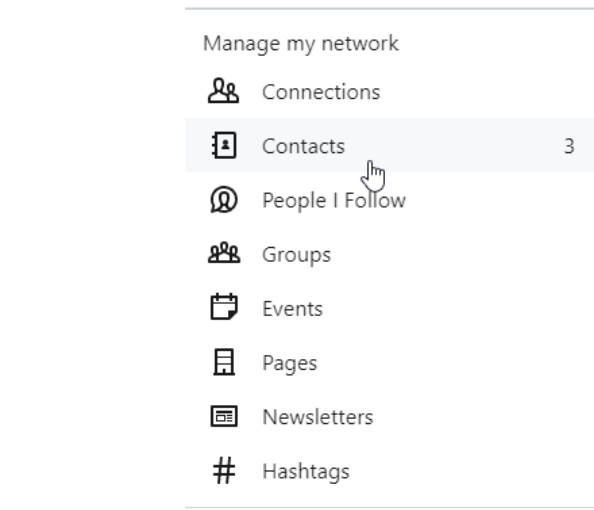


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>



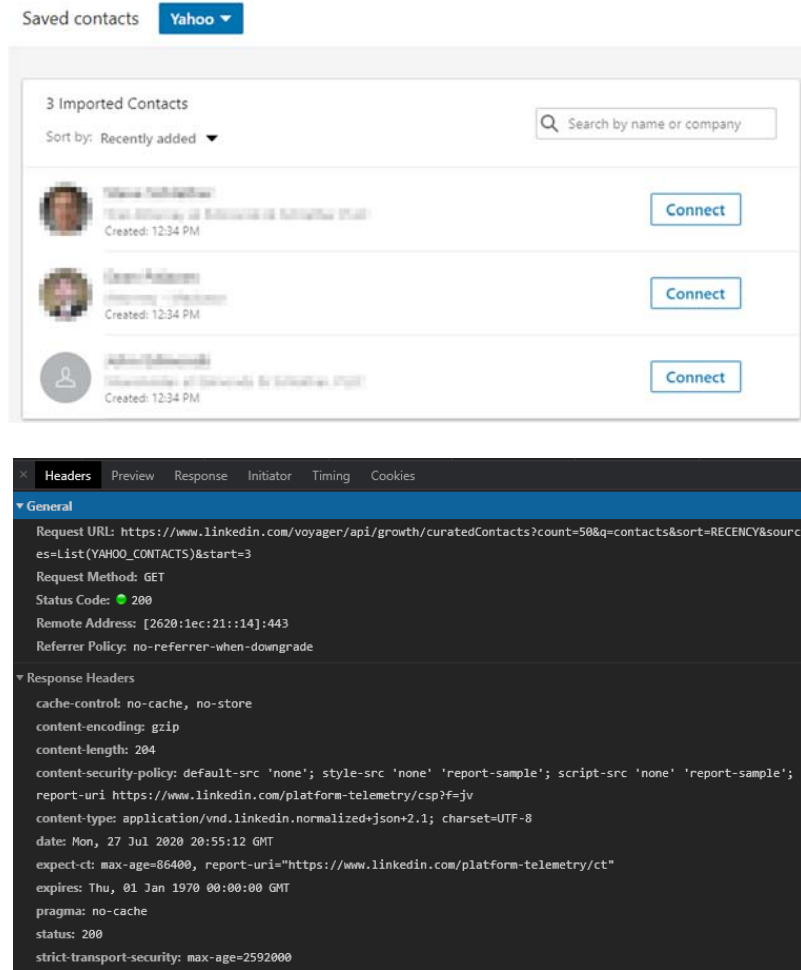


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

100. The LinkedIn Application comprises a network contacts database embodied in one or more non-transitory computer readable mediums. For example, the LinkedIn Application comprises an address book comprising at least the user's LinkedIn contacts embodied in, for example, a hard drive and/or other ROM and/or, at a minimum, RAM, for storing the user's contact list in said database:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

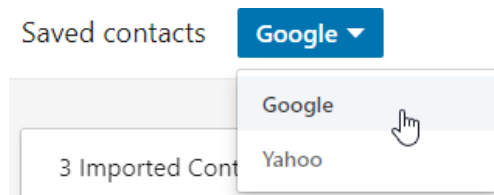
See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

Contact and Calendar Information

We receive personal data (including contact information) about you when others [import or sync their contacts or calendar with our Services](#), associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

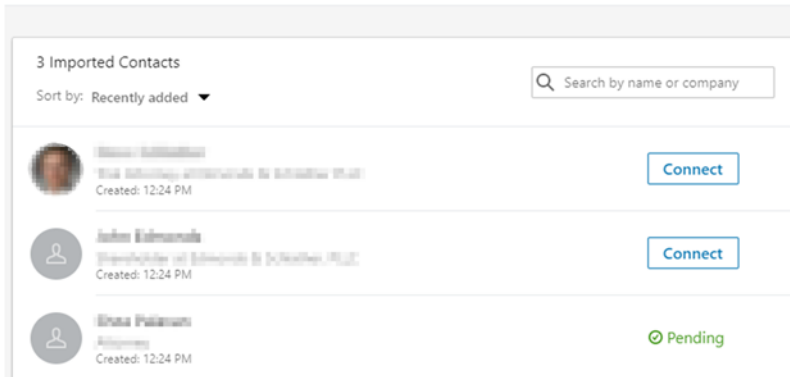
See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>





Saved contacts

Google ▼



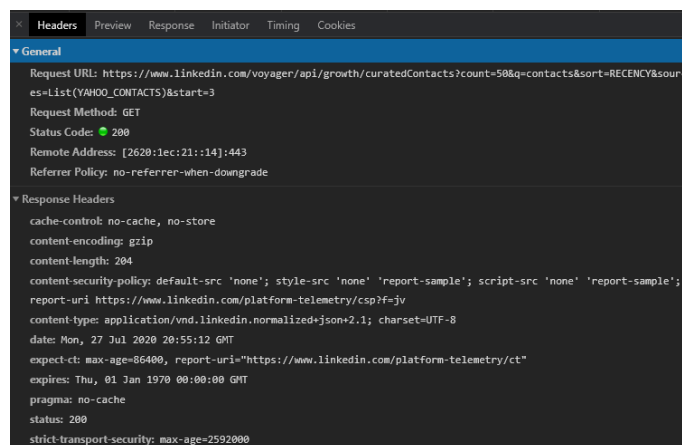
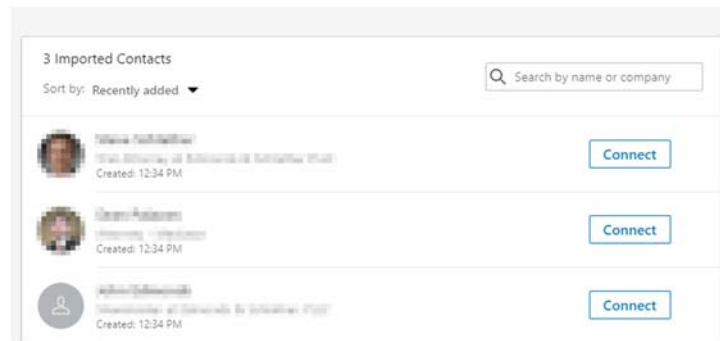
Saved contacts

Yahoo ▼



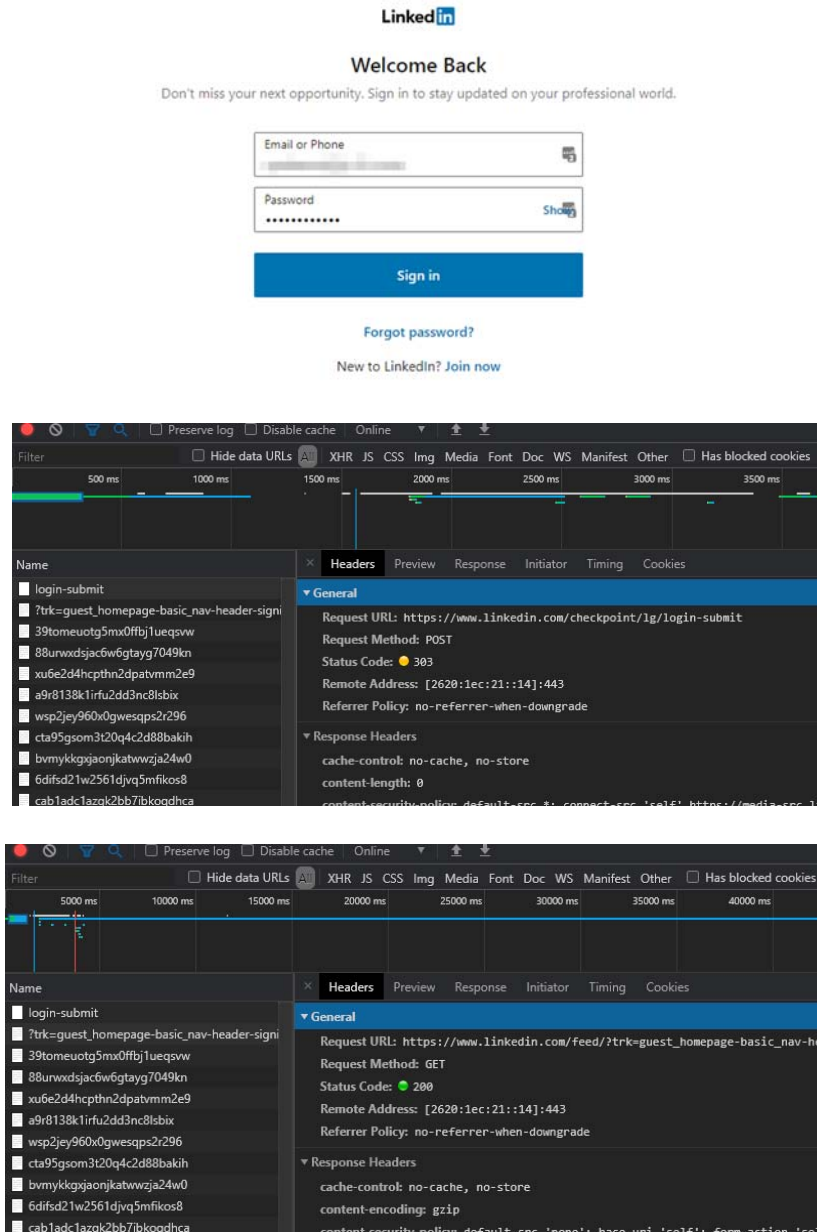
Saved contacts

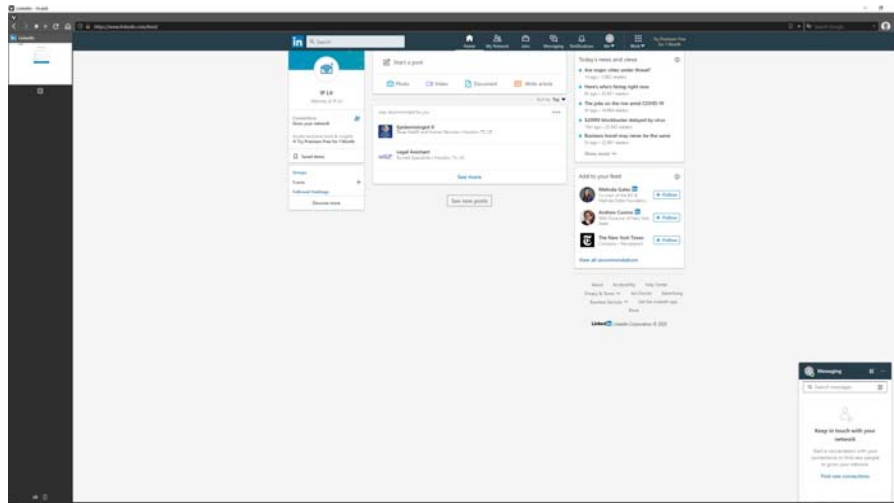
Yahoo ▼



See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

101. The LinkedIn Application comprises a server coupled to the network contacts database. For example, the LinkedIn Application comprises a server connected to the LinkedIn contact database:





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>

102. The LinkedIn Application comprises a contact aggregation engine coupled to the network login engine and the network contacts database. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager which provides LinkedIn's servers to aggregate the user's contacts coupled to the LinkedIn login engine and to the storage medium comprising the user's LinkedIn contacts:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your [contacts](#) and [connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically [import and store](#) details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect “email header” information that we can associate with Member profiles.

Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>

Syncing Contacts from Other Address Books and Sources

LinkedIn Contacts Manager can regularly synchronize with your contacts from Google Calendar and Google Contacts. Learn more about the **privacy of the information you sync**.

Note: If you're syncing a company email account, make sure you're in compliance with your corporate IT security policy first.

To sync contacts:

1. Click the **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click **Manage synced contacts** near the top right corner of the page.
4. Click **Sync** next to any source under the **Contacts** section to sync your contacts.

Notes:

- You will be prompted to login to the source account and give permission to process the sync.
- If you change your password for one of these sources, go back to this page and click **Change** to update it on LinkedIn.

To import a contacts file:

1. Click the **My Network** icon at the top of your LinkedIn homepage.
2. Click **Contacts** under **Manage my network** on the left rail.
3. Click **+ Add more contacts** on the right rail.

Note: You'll be redirected to a page where you can enter the source you want to import the contacts from.

Note: If you'd like to import a CSV file from a source that's not listed on the Contacts Settings page, import the file using the **Outlook Contacts CSV** option. This option is a workaround and may not work for all sources.

Important: Synced contacts are not automatically invited to connect with you on LinkedIn.

Learn more about **deleting imported contacts**, and why **old sources may be missing from the Contacts Syncing page**.

Last updated: 8 months ago

See, e.g., LinkedIn Help – Syncing Contacts from Other Address Books and Sources at <https://www.linkedin.com/help/linkedin/answer/1278>

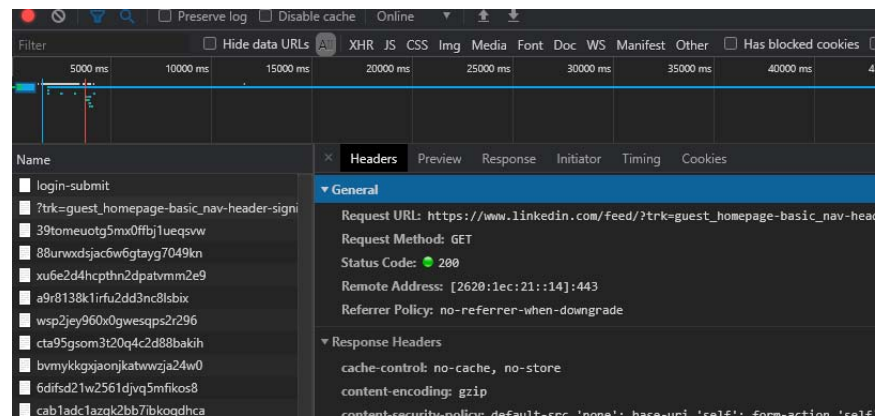
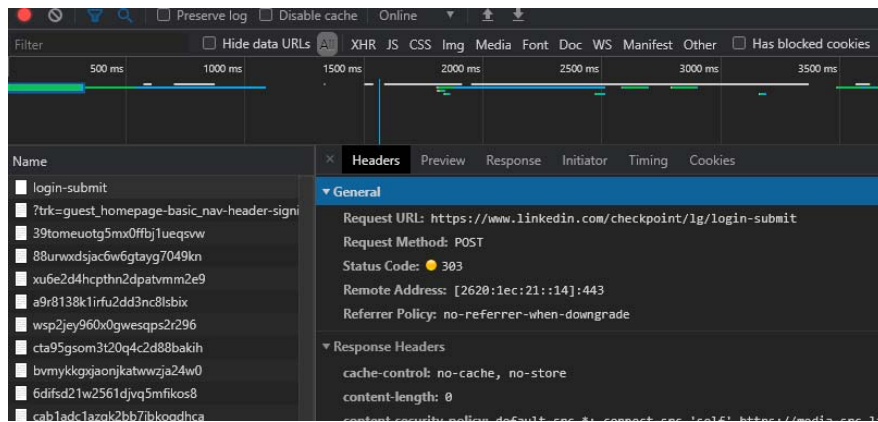
LinkedIn

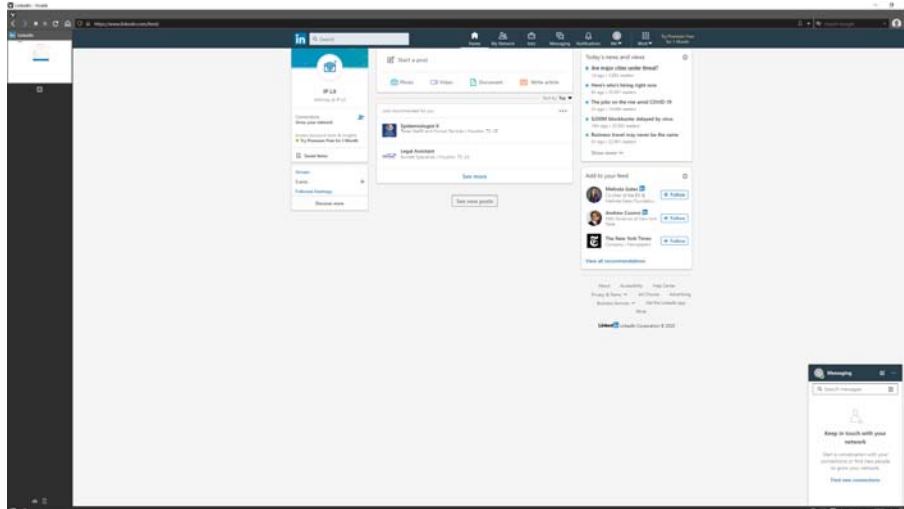
Welcome Back

Don't miss your next opportunity. Sign in to stay updated on your professional world.

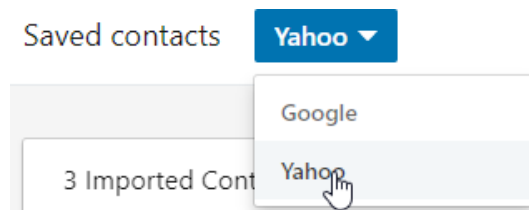
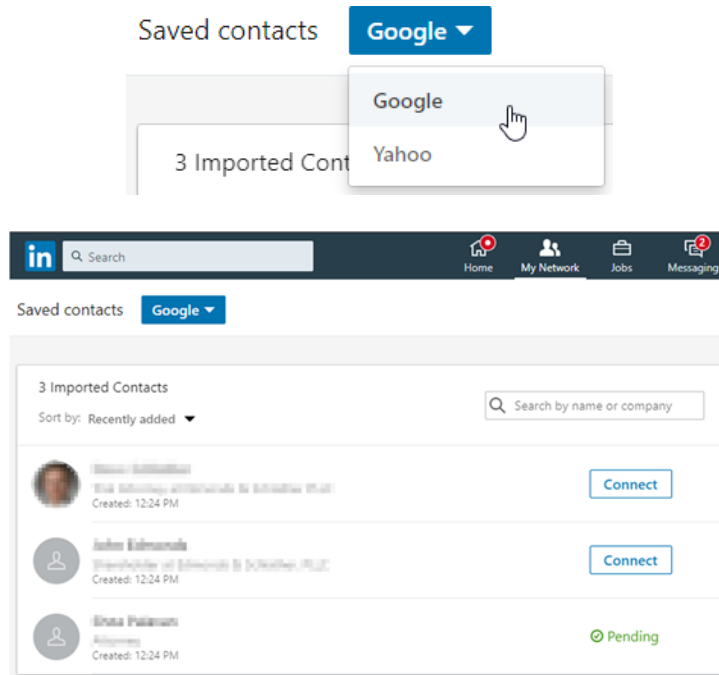
[Forgot password?](#)

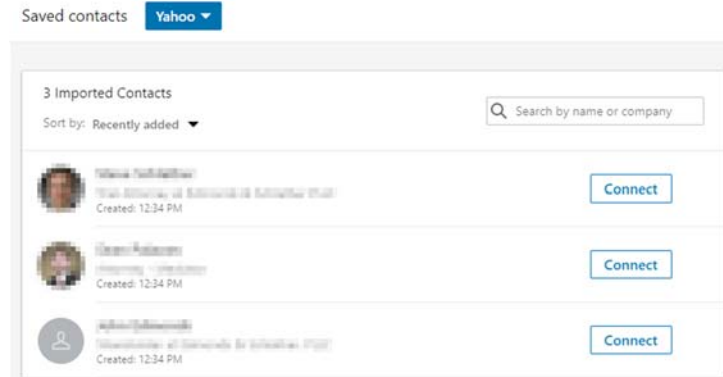
New to LinkedIn? [Join now](#)



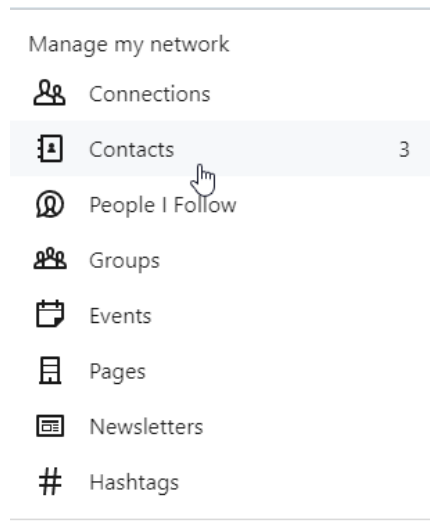


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com>

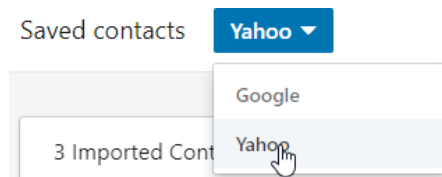


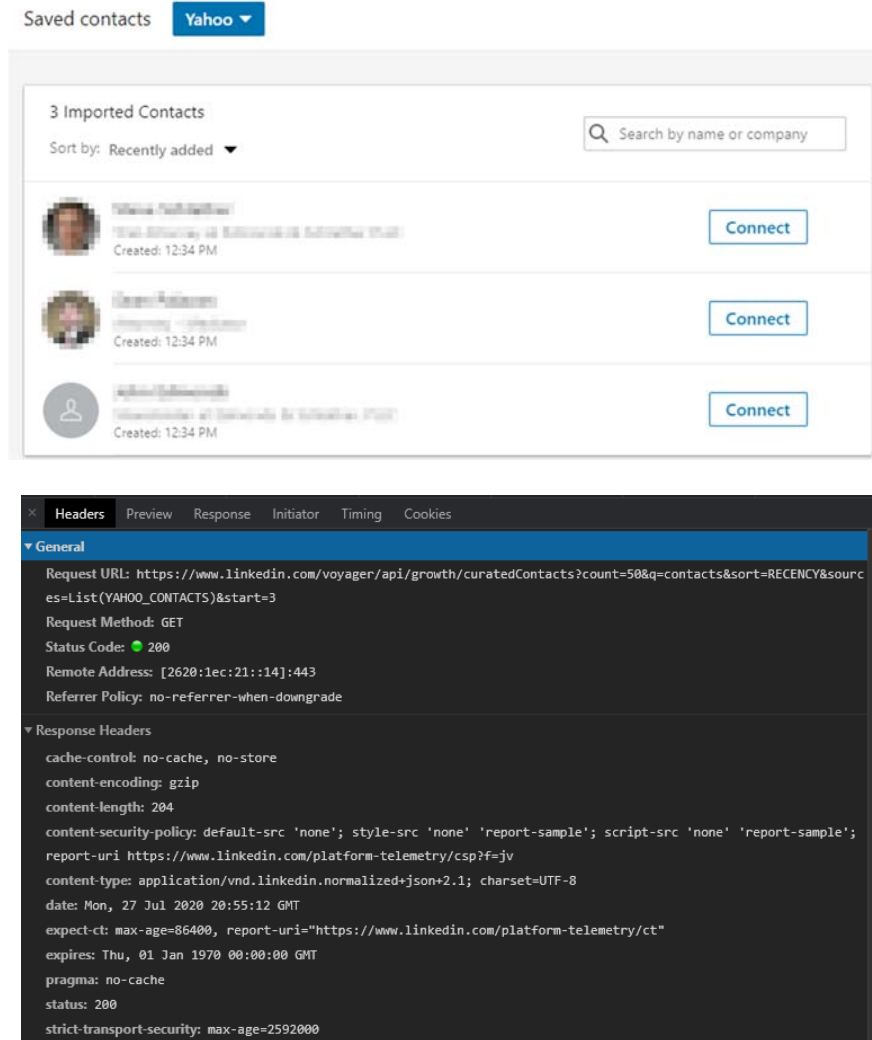


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



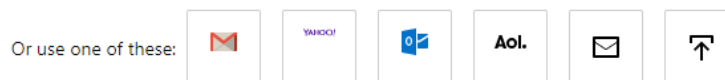
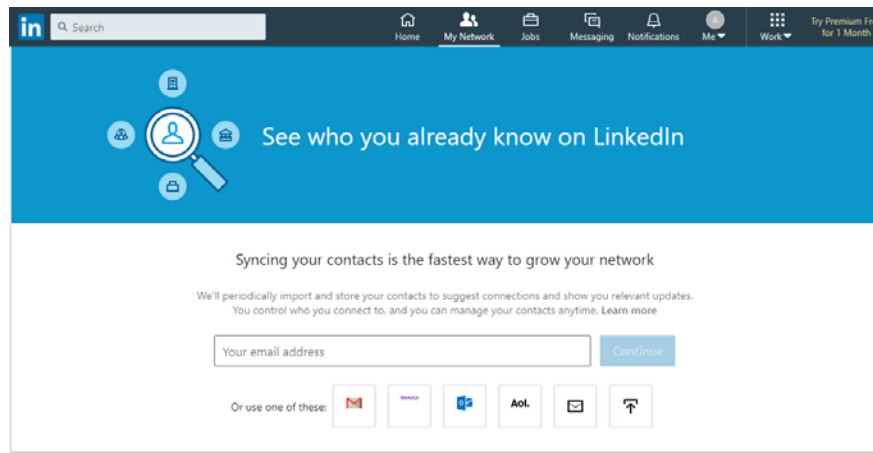
See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



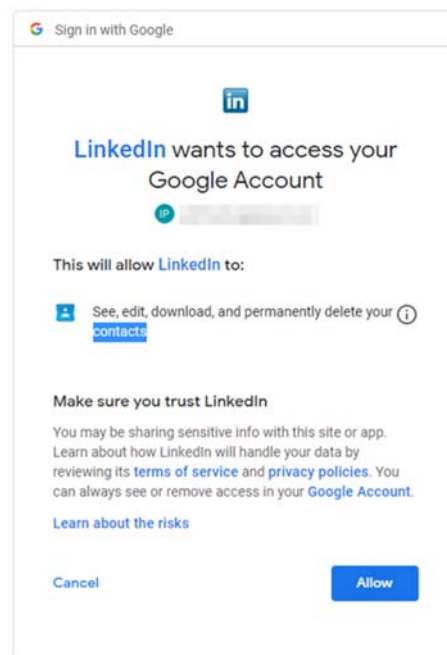


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

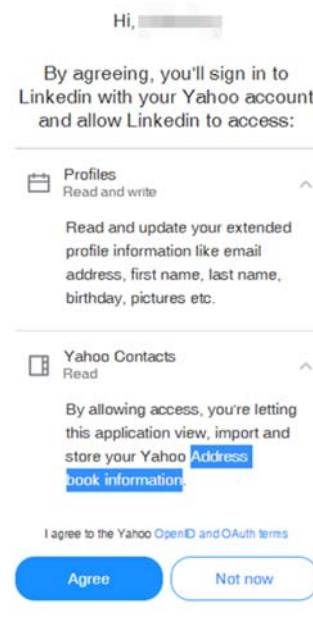
103. In operation, the contact aggregation engine (*see* above) controls the network login engine to login or facilitate login to a plurality of low level networks associated with a plurality of messaging service services through a high level network. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including facilitating the user logging into and associating the user's various social media and/or other networks, including via APIs of LinkedIn and/or APIs of the user's social media and/or other networks:



See, e.g., LinkedIn User Import Contacts page at <https://www.linkedin.com/mynetwork/import-contacts/?transactionId=>



See, e.g., Google Sign-In Request page at https://accounts.google.com/o/oauth2/v2/auth/identifier?client_id=802926523257.apps.googleusercontent.com&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=https%3A%2F%2Fwww.google.com%2Fm8%2Ffeeds%20profile%20email



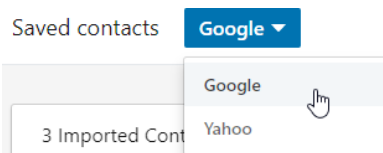
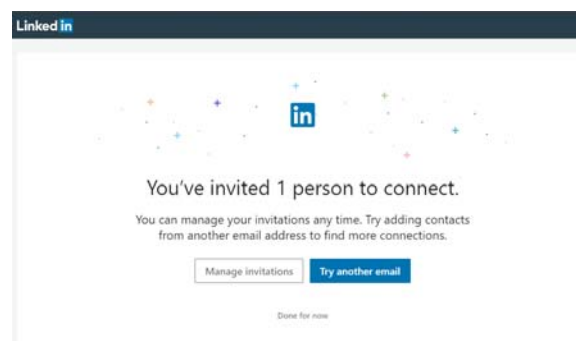
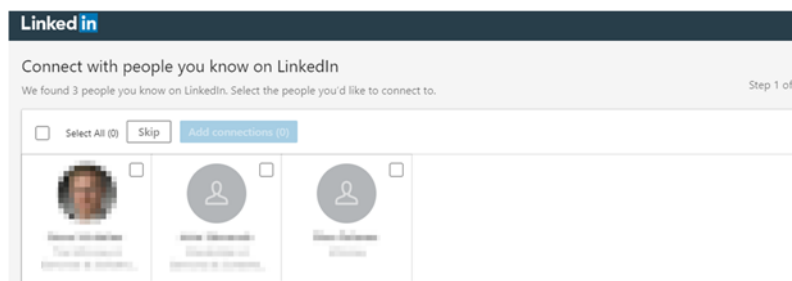
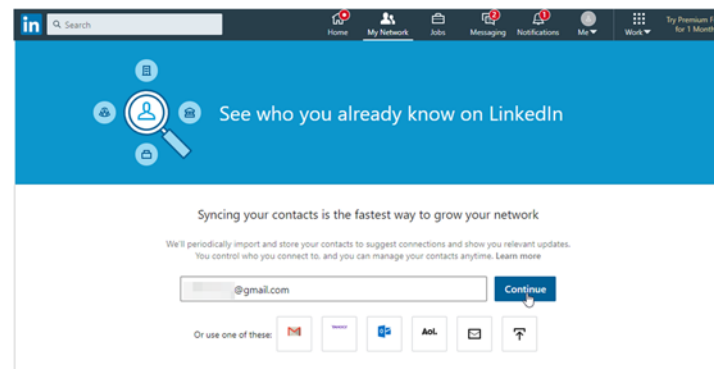
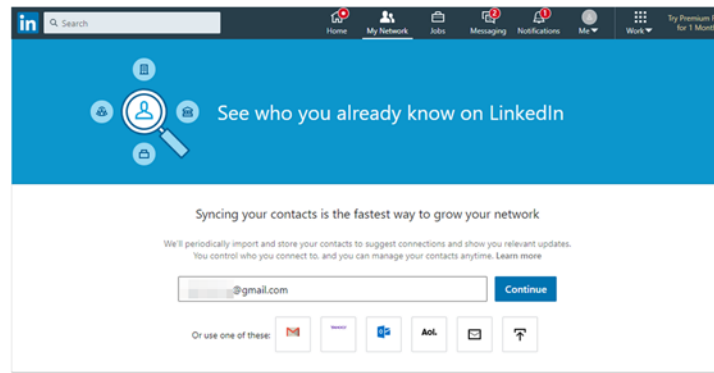
See, e.g., Yahoo! Sign-In Request page at

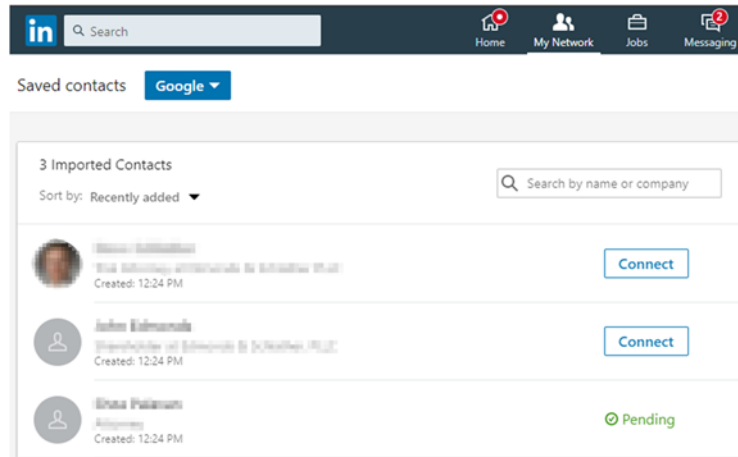
https://api.login.yahoo.com/oauth2/request_auth_fe?client_id=dj0yJmk9S0c3Y2RXVmw1RTQxJmQ9WVdrOVlYWmxXbWxDtXpBbWNHbzlnQS0tJnM9Y29uc3VtZXJzZWNyZXQmeD02MA--

&redirect_uri=https%3A%2F%2Fwww.linkedin.com%2Fgenie%2Ffinishauth&scope=openid%20sdpp-w%20sdct-r&sec=true&response_type=code&state=5ef8e091-7714-423c-a57b-f607710145fb&nonce=6612439201265914688&guccounter=1&guce_referrer=aHR0cHM6Ly9sb2dpbi55YWhvby5jb20v&guce_referrer_sig=AQAAADx0ox5D1DZWYJqN1T31M8wjeY72WHnm5USj7t1-

994XN4dtTs9wXZPxbFnm9LuyM26LTbfNgpUgeh89NDgcnS2xsBPJoIiDu6onlExDhf7tsoQ-EJKVkkqOu2ac4W3pj3P34o_G23xo8DLX4pMAqIKiP6olGZjJKhodFpcMc2yR

104. In operation, the contact aggregation engine (*see* above) uses the network interface to access contact information from the plurality of messaging services. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager obtaining and importing the contacts of the user's various added networks onto LinkedIn's servers via connection to the servers of these other networks, including via APIs of LinkedIn and/or these other networks, after the user logs in to the networks:





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

105. In operation, the contact aggregation engine (*see* above) updates the networks contacts database based on the contact information associated with the plurality of low level networks to create an aggregated contact list. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including creating and maintaining a combined, aggregated contact list using the user's imported contacts via the LinkedIn Contacts Manager, including maintaining a combined, aggregated contact list using the user's imported contacts via the LinkedIn Contacts Manager:

LinkedIn Contacts Manager - Overview

Bring together all the contacts from your address books, emails, calendars, and your LinkedIn network by using [LinkedIn Contacts Manager](#). Collecting and storing all your contacts in one place will help to keep them up to date.

Note: Your [contacts and connections](#) aren't the same thing.

You can begin using LinkedIn Contacts Manager by [importing and inviting your email contacts](#), [creating and uploading a contacts file](#), and [syncing contacts from other sources](#). Currently, you can only sync Google Calendar and Google Contacts, so [old sources may be missing](#).

Once you've synced and imported your contacts, you can [delete specific contacts](#) to remove them from your [LinkedIn Contacts Manager address book](#). If you've accidentally sent invitations to the contacts you imported, you can [withdraw the invitations](#).

LinkedIn takes your privacy and the security of your data seriously. Learn more about the [privacy of your information in LinkedIn Contacts Manager](#), and how you can [access your account data](#).

Last updated: 7 months ago

See, e.g., LinkedIn Help – LinkedIn Contacts Manager – Overview at <https://www.linkedin.com/help/linkedin/answer/91972>

Viewing and Managing Your Email Contacts

If you've enabled address book syncing, we'll periodically **import and store** details about your address book contacts to suggest relevant contacts for you to connect with, to show you relevant updates, and for other uses explained in our [Privacy Policy](#).

We save the contact data returned by your email provider. This could include names, birthdays, gender, locations, job titles, email addresses, phone numbers, websites, and notes.

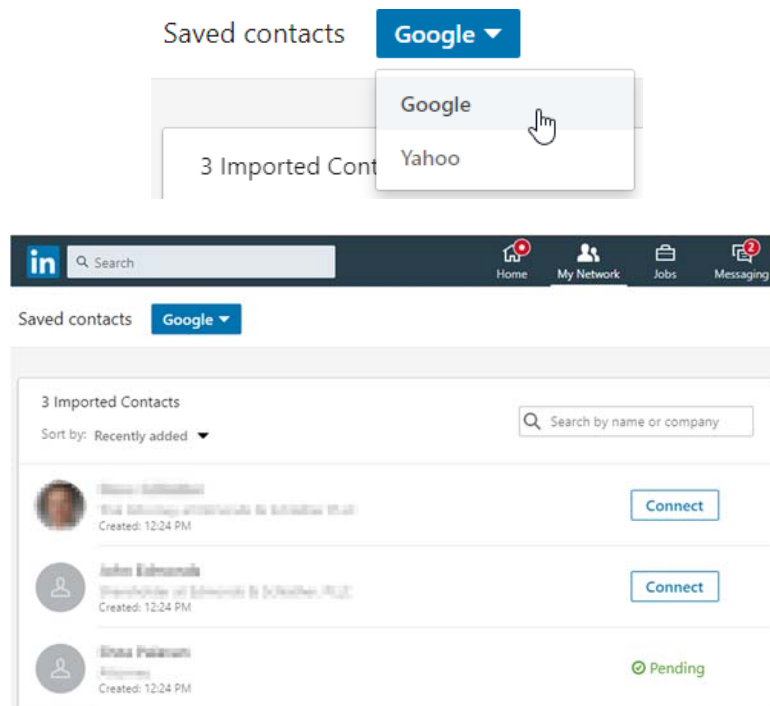
See, e.g., LinkedIn Help – Viewing and Managing Your Email Contacts at <https://www.linkedin.com/help/linkedin/answer/98247>

Contact and Calendar Information

We receive personal data (including contact information) about you when others **import or sync their contacts or calendar with our Services**, associate their contacts with Member profiles, scan and upload business cards, or send messages using our Services (including invites or connection requests). If you or others opt-in to sync email accounts with our Services, we will also collect "email header" information that we can associate with Member profiles.

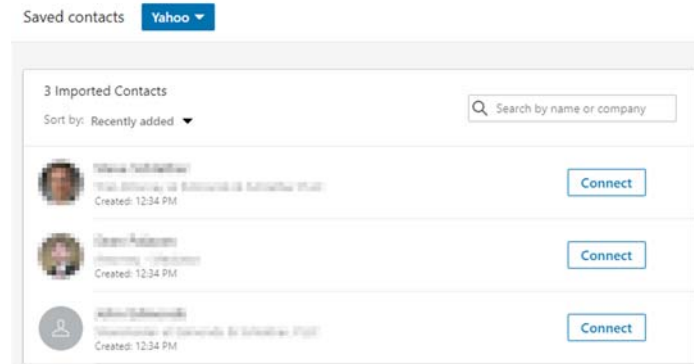
Others may sync their contacts or calendar with our Services.

See, e.g., LinkedIn Privacy Policy at <https://www.linkedin.com/legal/privacy-policy>



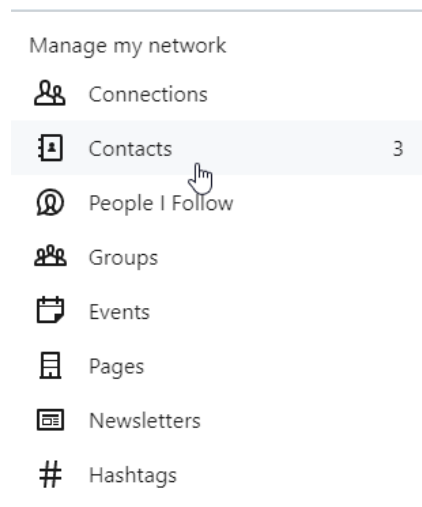
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>



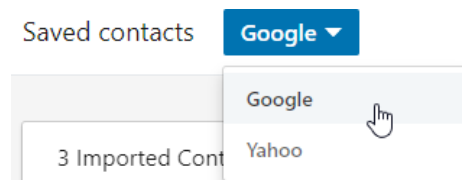


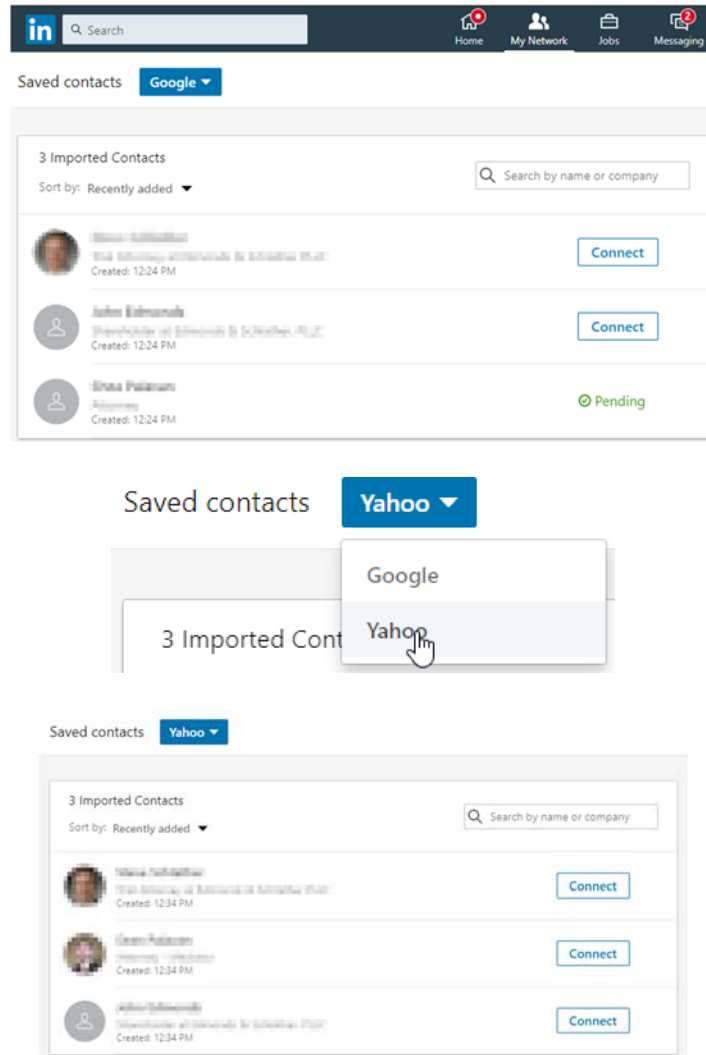
See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

106. In operation, the contact aggregation engine (*see* above) stores the aggregated contact list in a non-transitory computer-readable medium at the server. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including storing the user's aggregated contact list on LinkedIn's servers:

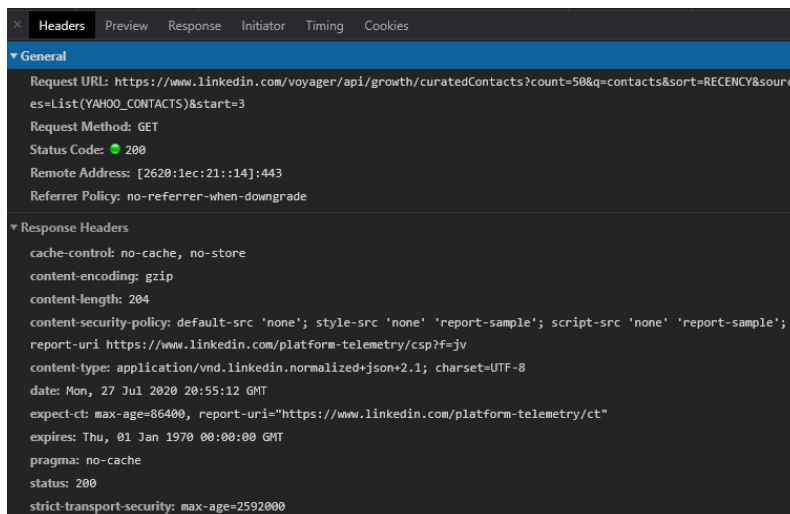


See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>



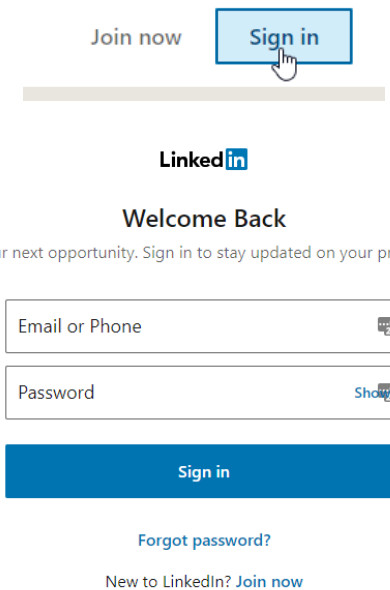


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

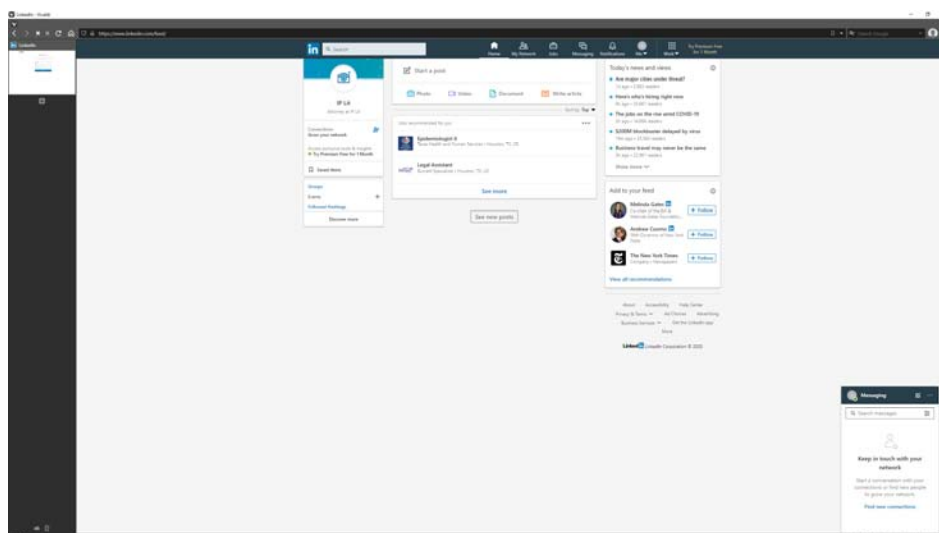


See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>

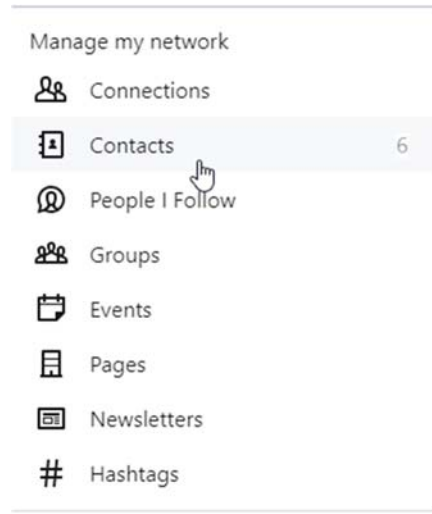
107. In operation, the contact aggregation engine (*see* above) provides the aggregated contact list including the contact information to a display device. For example, the LinkedIn Application comprises the LinkedIn Contacts Manager that performs steps while in operation for, *inter alia*, facilitating aggregation of contacts from multiple messaging service providers on multiple networks associated therewith, including providing the user's aggregated contact list stored on LinkedIn's servers, which comprises the user's LinkedIn contacts and imported contacts:



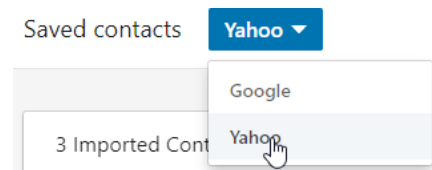
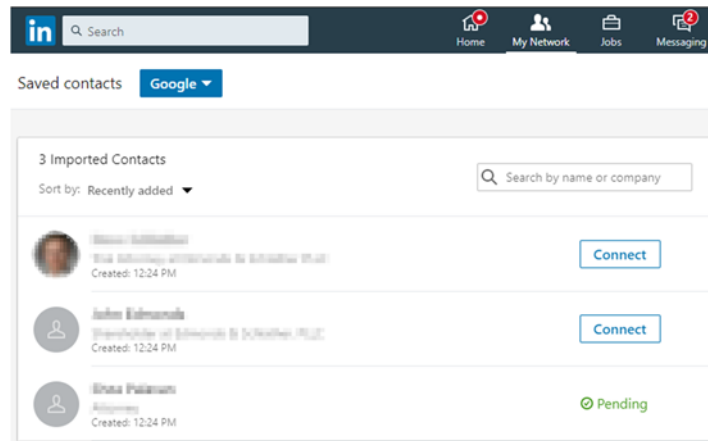
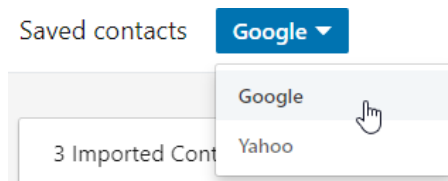
The image shows the LinkedIn login interface. At the top, there are two buttons: "Join now" and "Sign in". A hand cursor is pointing at the "Sign in" button. Below these buttons is the LinkedIn logo. Under the logo, it says "Welcome Back" and "Don't miss your next opportunity. Sign in to stay updated on your professional world." There are two input fields: "Email or Phone" and "Password". The "Password" field has a "Show" link next to it. Below the input fields is a large blue "Sign in" button. Underneath the button are two links: "Forgot password?" and "New to LinkedIn? Join now".

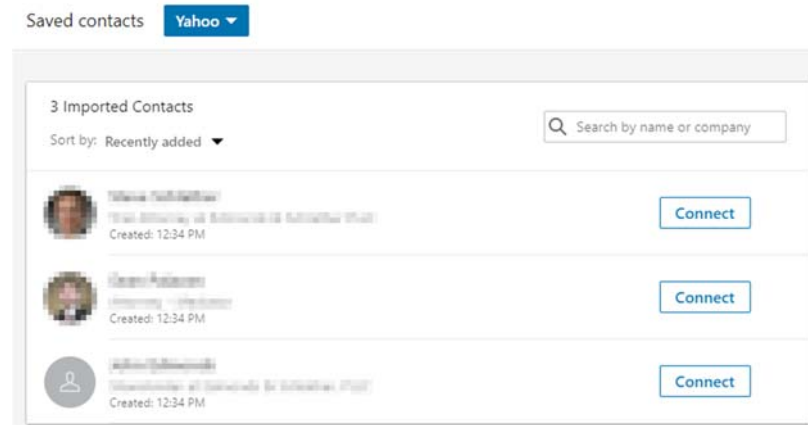


See, e.g., LinkedIn Login page at <https://www.linkedin.com/login>



See, e.g., LinkedIn User MyNetwork page at <https://www.linkedin.com/mynetwork/>





See, e.g., LinkedIn Desktop Website at <https://www.linkedin.com/mynetwork/>.

108. LinkedIn has directly infringed, and continues to directly infringe, the claims of the ‘453 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

109. Additionally, or in the alternative, LinkedIn has induced, and continues to induce, infringement of the ‘453 Patent in this judicial district, and elsewhere, by actively inducing direct infringement of the ‘453 Patent, including by knowingly and actively aiding or abetting infringement by users, by and through at least instructing and encouraging the use of the LinkedIn products and software noted above, including the LinkedIn Application system. Such aiding and abetting comprises providing software, web servers, and/or instructions regarding the use and/or operation of the LinkedIn Application system, applications, and web servers in an infringing manner. Further, the direct infringement of users that occurs in connection with LinkedIn’s applications and/or web services occurs under the direction or control of LinkedIn. Such induced infringement has occurred since LinkedIn became aware of the ‘453 Patent, at a minimum, as noted above, and the knowledge and awareness that such actions by users comprise infringement of the ‘453 Patent.

110. LinkedIn has had at least constructive notice of the ‘453 Patent since at least its issuance. LinkedIn will have been on actual notice of the ‘453 Patent since, at the latest, the service

of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the '453 Patent.

111. The LinkedIn Application system clearly meets the asserted claim limitations in their normal and expected usage. On information and belief, normal and expected usage of the LinkedIn Application system by customers and/or end users satisfies the claim limitations for direct infringement. Further, at minimum, the provision of products, systems and/or functionalities clearly capable of such infringing usage and/or provision of instructions/specifications for such infringing usage constitutes inducement of directly infringing usage.

112. Further, as noted above, LinkedIn was made aware of infringement of the '453 patent through use of the LinkedIn Application via the infringement allegations set forth in Plaintiff's Original Complaint, of which Defendants were aware at least as of the service of said Original Complaint. Such direct and induced infringement has been and remains clear, unmistakable and inexcusable. On information and belief, LinkedIn knew or should have known of the clear, unmistakable and inexcusable direct and induced infringing conduct at least since receiving notice of the '453 Patent. Thus, on information and belief, Defendants have, since receiving notice of the '453 Patent, specifically intended to induce direct infringement by customers and/or end users.

113. EBT believes and contends that, at minimum, LinkedIn's knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the '453 Patent since receiving notice of its infringement of the '453 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn

has willfully infringed the ‘453 Patent.

COUNT III – INFRINGEMENT OF U.S. PATENT NO. 8,230,135

114. The ‘135 Patent is entitled “Event Notification System And Method.” The application for the ‘135 Patent was filed on June 21, 2011. The patent issued on July 24, 2012. The ‘135 Patent claims priority to provisional and non-provisional filings dated as far back as December 9, 2005.

115. Plaintiff is the owner of the ‘135 Patent and it has all substantial rights to the ‘135 Patent, including the right and standing to sue and recover damages for past, present and future infringement of the patent.

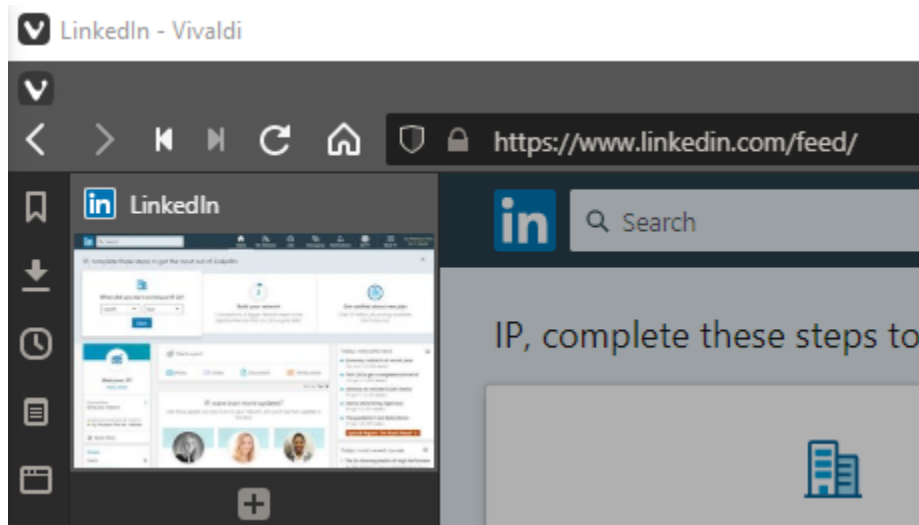
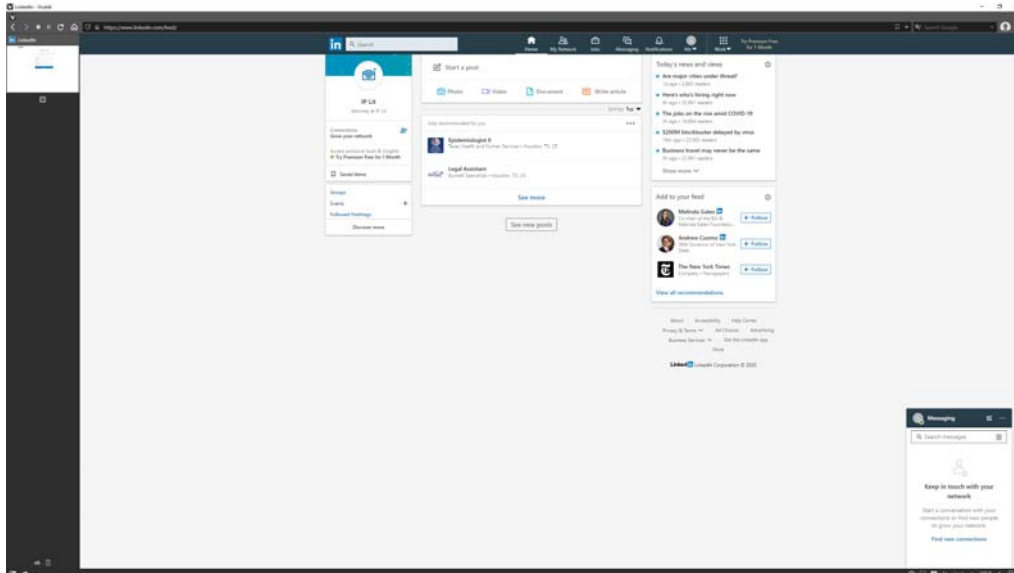
116. Claim 1 of the ‘135 Patent covers a method comprising “receiving information of an event that calls for user notification; generating an event notification for the event; associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; providing the at least one of the plurality of character strings in the title array to a process executed by a processor; providing an alternative title based on the at least one of the plurality of character strings to the process; using the alternative title as a title in association with the process.”

117. Defendant has infringed, and is now infringing, the ‘135 patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing of methods and/or providing of systems, without authority from Plaintiff, for notifying a user of the occurrence of events and notification apparatuses and functionality, including notifying a user of the occurrence of an event by modification of the user’s browser title bar based on the specific event that has occurred, including as claimed in the ‘135

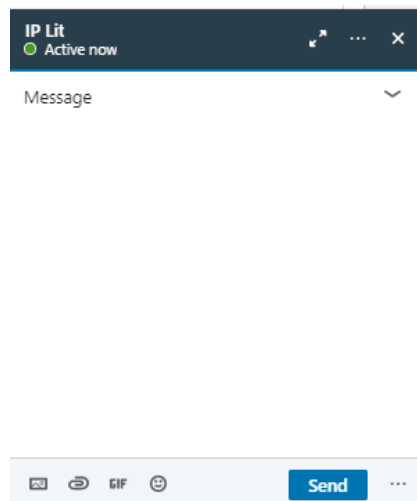
asserted claims. On information and belief, Defendant practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising www.linkedin.com.

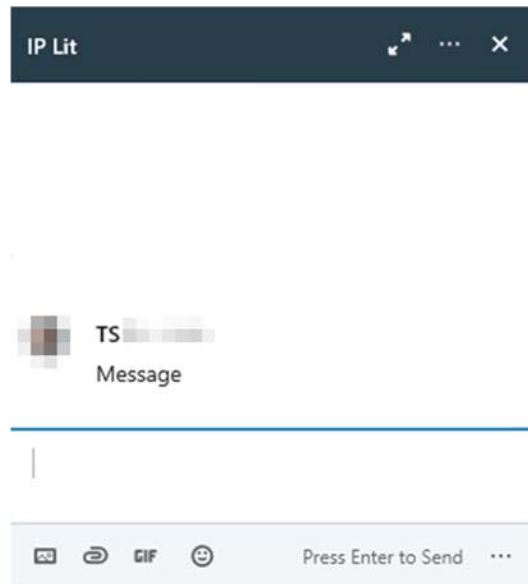
118. Without limitation, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods comprises receiving information of an event that calls for user notification; generating an event notification for the event; associating the event notification with at least one of the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device; providing the at least one of the plurality of character strings in the title array to a process executed by a processor; providing an alternative title based on the at least one of the plurality of character strings to the process; using the alternative title as a title in association with the process. For example, the LinkedIn Application system permits a user's device to notify the user, via the user's browser, that the user has notifications for an event, such as a new message and/or other posting on the LinkedIn website, including via updating, modifying, and/or otherwise altering the title bar of the user's browser, including instructing the user's browser to receive notifications regarding messages the user has received from other LinkedIn users and/or posting of other LinkedIn users.

119. The LinkedIn Application receives information of an event that calls for user notification. For example, the LinkedIn Application receives information of an event that calls for user notification and it instructs and/or controls the user's browser on the user's device running code which receives code or other instructions from LinkedIn that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>





See, e.g., Second LinkedIn User Feed page at <https://www.linkedin.com/feed>

[illegible]

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask(" _loopTitles","".concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%N] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}}
23413 e.default=n))
23414 define("image-edit-base/components/bg-img-edit" ["exports" "image-edit-base/components/picture-cropper" "image-edit

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 if(!null!=a) return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new l.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)("latestEvent") ill==a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 if(void 0!==s&&q.has(s)){o = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 return n} if(void 0!==(o=e[t])){"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 if(!null!=a) return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new l.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)("event") s=null==a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 if(void 0!==s&&q.has(s)){o = u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 return n} if(void 0!==(o=e[t])){"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)

```

```

2112 void 0===n-e[t]&&(!0|t in e||function i!=typeof e.setUnknownProperty){n=e.setUnknownProperty(t,n)}
2113 if(a&&ee()){Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray(n))&&Z(ie(n,"[]"));0,r.isProxy(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(!null!=a) return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])){"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))|this._volatile=!1
2124 n.readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a="function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===n?e[t]&&(!0|t in e||"function"!=typeof e.setUnknownProperty){n=e.setUnknownProperty(t)}
2113 if(a&&ee()){Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray(n))&&Z(ie(n,"[]"));0,r.isProxy(n)&&Z(ie(n,"content"))return n}functi
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(!null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0!==(o=e[t])){"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))|this._volatile=!1
2124 n.readOnly=!1

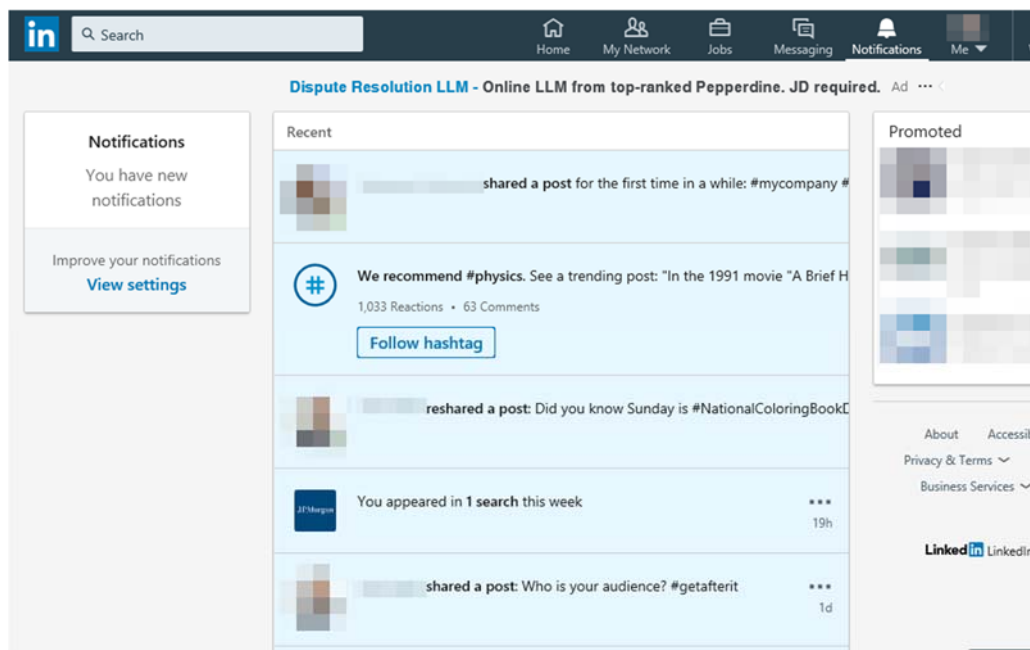
```

```

MessageEvent
isTrusted: false
lastEventId: "null"
origin: ""
▶ path: []
▶ ports: []
returnValue: true
source: null
srcElement: null
target: null
timeStamp: 6400162.895000001
type: "message"
userActivation: null
__proto__: MessageEvent

```

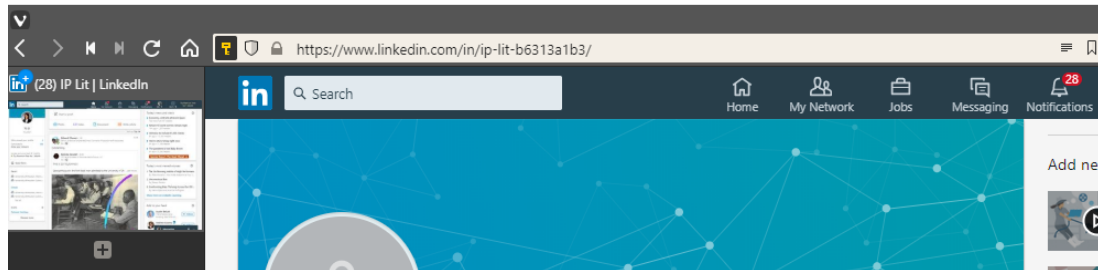
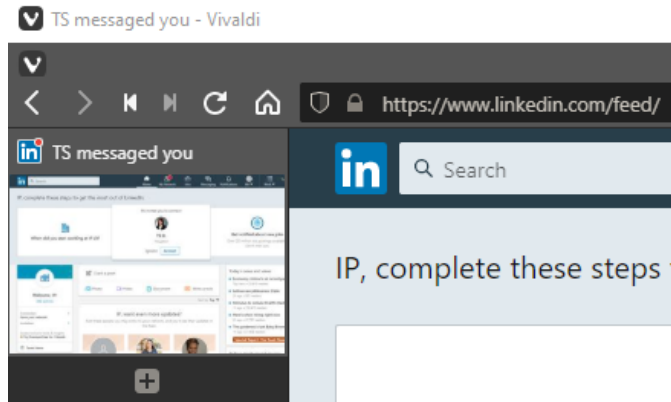
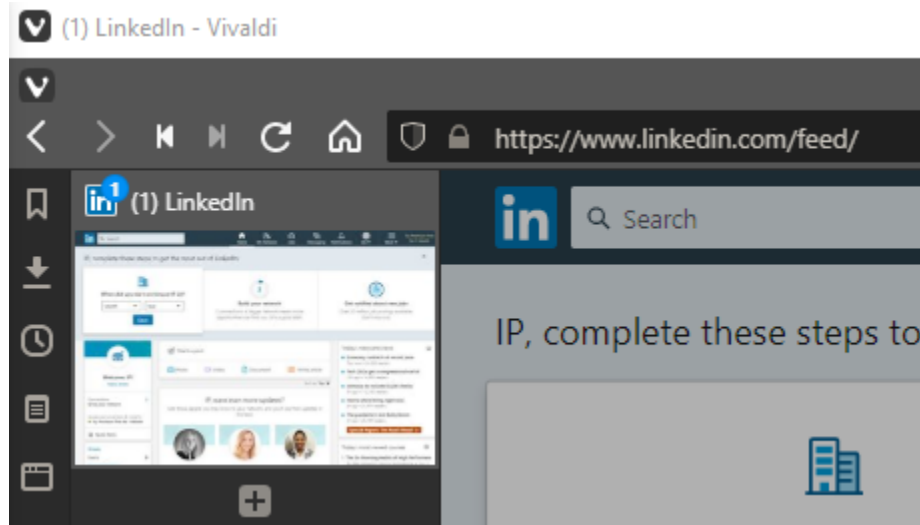
See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>



See, e.g., LinkedIn User Notification page at <https://www.linkedin.com/notifications>

120. The LinkedIn Application generates an event notification for the event. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which generates a notification to indicate to the user that the user has received a

message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle=""},addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this._pollTask("._loopTitles","".concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e,getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}})
23413 e.default=n))
23414 define("image-edit-base/components/bg-image-edit" ["exports", "image-edit-base/components/picture-cropper" "image-edit

```

```

2112 void 0==(n=e[t])&&(o||t in e||function i=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&eee())Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)●return Ne(a,o,n), o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o||"function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(o||t in e||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&eee())Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))return n}functi
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)●return Ne(a,o,n), o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.')}}(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

121. The LinkedIn Application associates the plurality of character strings in a title array that includes a plurality of character strings for provisioning for display in a titlebar or taskbar of a display device. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings and inputs the created notification into an array of title strings which are specifically for use by the user's browser to display in the titlebar of the user's browser which is displayed on the display of the user's device:

```

define("html-document-title/services/document-title-poller",{"exports":{
  "use-strict":true
},Object.defineProperty(e,"__esModule",{
  value:true
}),{
  default:void 0
},var n=Ember.Service.extend({
  init:function(){
    this._super.apply(this,arguments)
    this._titles=[]
    this._loopNumber=0
    this._defaultDocumentTitle=""
  },
  addTitle:function(e){
    if(!this._titles.length&&t.default){
      this.updateDefaultTitle(document.title)
      this._titles=[e,this._defaultDocumentTitle]
    }else{
      this._titles.unshift(e)
      this._titles=this._titles.uniq()
    }
    if(!this._pollerEnabled){
      this._pollerEnabled=true
      this.pollTask("_loopTitles","").concat("docume
    },
  },

```



```

8415 define("ember-lifeline/poll-task", ["exports", "ember-lifeline/utils/get-task", "ember-lifeline/utils/disposable"],
8416   "use strict"
8417   Object.defineProperty(e, "__esModule", {
8418     value: !0
8419   })
8420   e._setRegisteredPollers = function(e) {
8421     a = e
8422   }
8423   e.setShouldPoll = function(e) {
8424     o = e
8425   }
8426   e.pollTask = function(e, r) {
8427     var c, m = arguments.length > 2 && void 0 !== arguments[2] ? arguments[2] : !(), u = (0,
8428     t.default)(e, r, "pollTask"), d = function() {
8429       return u.call(e, c)
8430     }, p = a.get(e)
8431     if (!p) {
8432       p = new Set
8433       a.set(e, p);
8434       (0,
8435       n.registerDisposable)(e, (function(e, t) {
8436         return function() {
8437           t.forEach((function(t) {
8438             s(e, t)
8439           )
8440         })
8441       )
8442       )(e, p))
8443     }
8444     p.add(m)
8445     c = (function() {
8446       if (o)
8447         return o()
8448       return !ember.testing
8449     })() ? d : function() {
8450       i[m] = d
8451     }
8452     u.call(e, c)
8453     return m
8454   }
8455   e.cancelPoll = s
8456   e.queuedPollTasks = void 0
8457   var a = new WeakMap
8458   var o, r = 0
8459   var i = Object.create(null)
8460   e.queuedPollTasks = i
8461   function s(e, t) {
8462     var n
8463     if ("number" == typeof e || "string" == typeof e)
8464       n = e
8465     else {
8466       var o = a.get(e)
8467       n = t
8468       void 0 !== o && o.delete(n)
8469     }
8470     delete i[n]
8471   }
8472   function l() {
8473     return r++
8474   }
8475   })
8476   })
8477   })

```

```

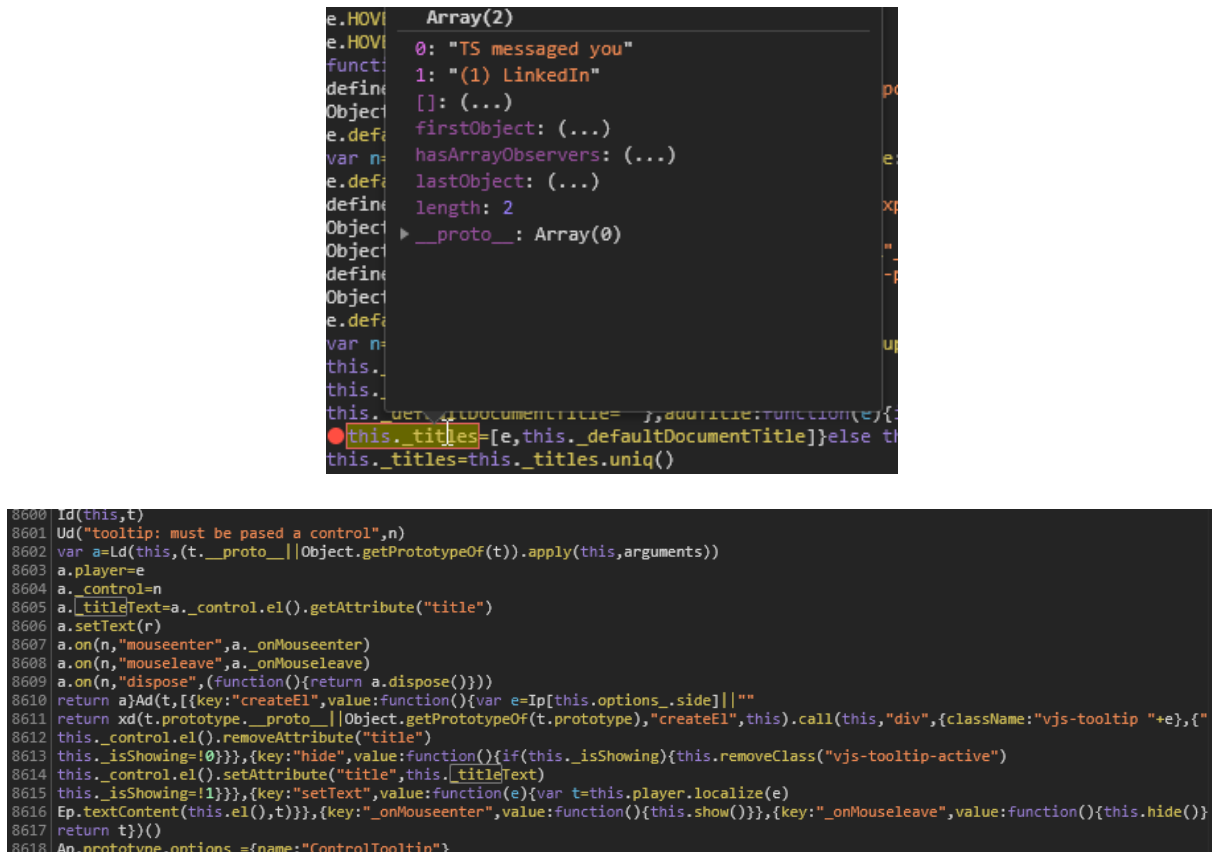
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle=""},addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("_loopTitles","".concat("document-title-poller","").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e,getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}})
23413 e.default=n))
23414 define("image-edit-base/components/bg-img-edit" ["exports", "image-edit-base/components/picture-cropper", "image-edit

```

```

2112 void 0==(n=e[t])&&(!t in e||function t=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&e){Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!ie.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)return Ne(a,o,n,o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&s.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&ie(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

122. The LinkedIn Application provides the at least one of the plurality of character strings in the title array to a process executed by a processor. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the entries in the titlebar array for use by the code on the user's browser which is run by the processor of the user's device:

An Introduction to JavaScript

Let's see what's so special about JavaScript, what we can achieve with it, and which other technologies play well with it.

What is JavaScript?

JavaScript was initially created to "make web pages alive".

The programs in this language are called *scripts*. They can be written right in a web page's HTML and run automatically as the page loads.

Scripts are provided and executed as plain text. They don't need special preparation or compilation to run.

In this aspect, JavaScript is very different from another language called *Java*.

Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called the JavaScript engine.

The browser has an embedded engine sometimes called a "JavaScript virtual machine".

Different engines have different "codenames". For example:

- V8 – in Chrome and Opera.
- SpiderMonkey – in Firefox.
- ...There are other codenames like "Trident" and "Chakra" for different versions of IE, "ChakraCore" for Microsoft Edge, "Nitro" and "SquirrelFish" for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We'll use them too. For instance, if "a feature X is supported by V8", then it probably works in Chrome and Opera.

How do engines work?

Engines are complicated. But the basics are easy.

1. The engine (embedded if it's a browser) reads ("parses") the script.
2. Then it converts ("compiles") the script to the machine language.
3. And then the machine code runs, pretty fast.

The engine applies optimizations at each step of the process. It even watches the compiled script as it runs, analyzes the data that flows through it, and further optimizes the machine code based on that knowledge.

What can in-browser JavaScript do?

Modern JavaScript is a "safe" programming language. It does not provide low-level access to memory or CPU, because it was initially created for browsers which do not require it.

JavaScript's capabilities greatly depend on the environment it's running in. For instance, [Node.js](#) supports functions that allow JavaScript to read/write arbitrary files, perform network requests, etc.

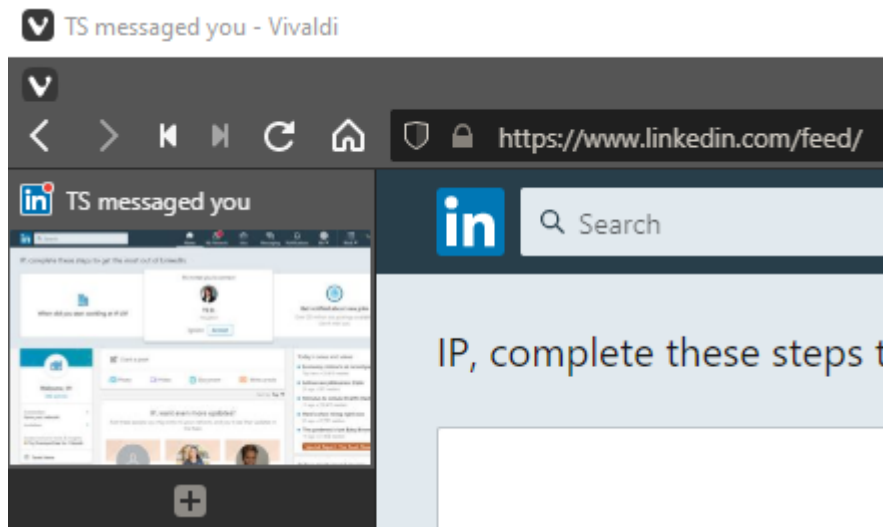
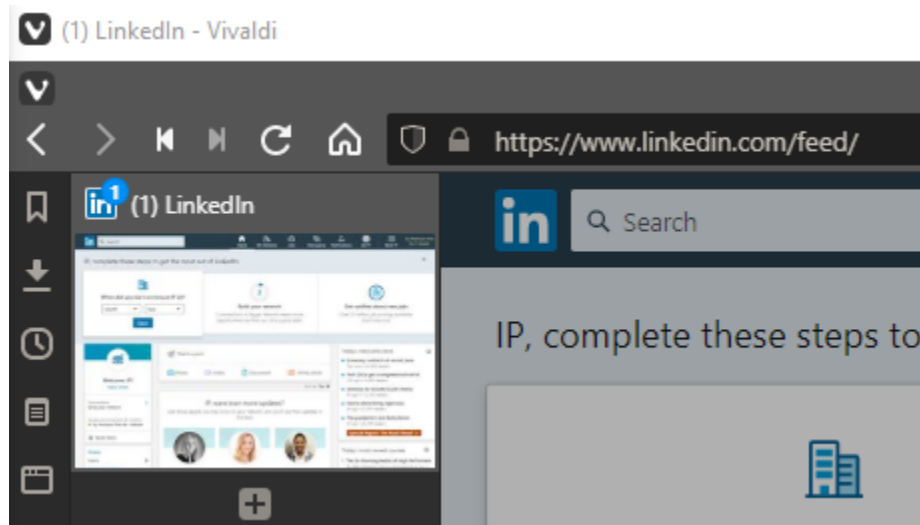
In-browser JavaScript can do everything related to webpage manipulation, interaction with the user, and the webserver.

For instance, in-browser JavaScript is able to:

- Add new HTML to the page, change the existing content, modify styles.
- React to user actions, run on mouse clicks, pointer movements, key presses.
- Send requests over the network to remote servers, download and upload files (so-called [AJAX](#) and [COMET](#) technologies).
- Get and set cookies, ask questions to the visitor, show messages.
- Remember the data on the client-side ("local storage").

See, e.g., An Introduction to JavaScript on JavaScript.info at <https://JavaScript.info/intro>

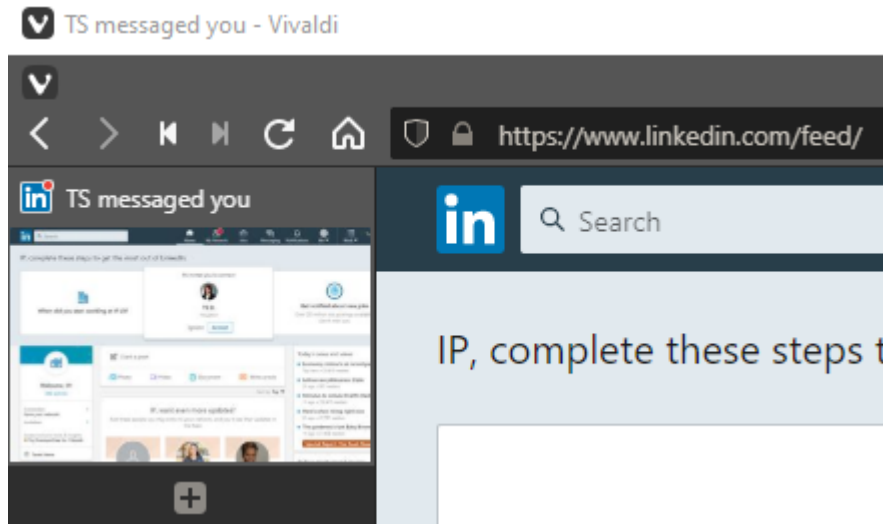
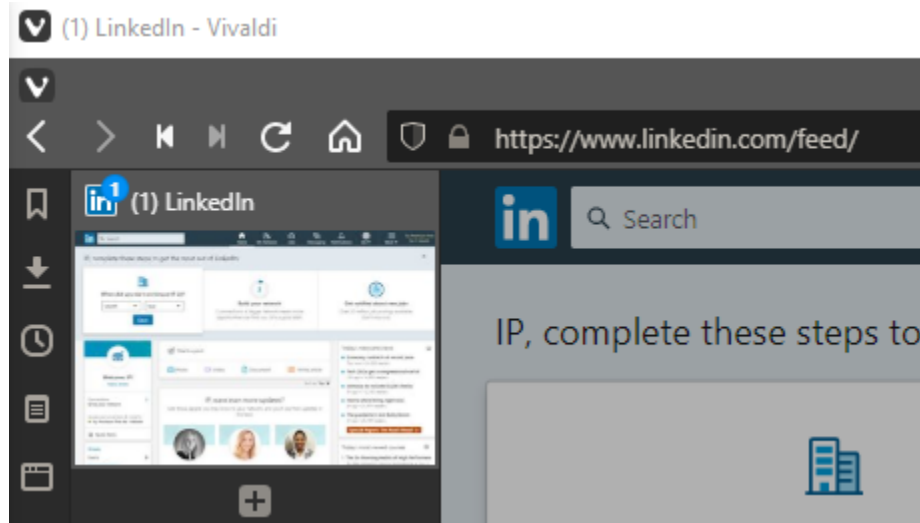
123. The LinkedIn Application provides an alternative title based on the at least one of the plurality of character strings to the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in the titlebar, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

124. The LinkedIn Application uses the alternative title as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which causes the user's browser to use the alternative title or titles as

the titlebar for the user's browser by updating the HTML run and displayed by the user's browser:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask(" _loopTitles","".concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e,getDefaulTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}})
23413 e.default=n)
23414 define("image-edit-base/components/hq-img-edit" ["exports" "image-edit-base/components/picture-cropper" "image-edit

```

```

2112 void 0==(n=e[t])&&(!o||t in e||function t!=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&e(e)){Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray)(n)&&Z(ie(n,"[]"));0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n;if(void 0!==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){0,t.inheritsLoose}(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o||"function"===i
2111 if(Ae(t))return aPe(e,t);void 0
2112 void 0==(n=e[t])&&(!o||t in e||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&e(e)){Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray)(n)&&Z(ie(n,"[]"));0,r.isProxy)(n)&&Z(ie(n,"content"))return n}functi
2114 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e =
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "'+i.join(".")+'" could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n;if(void 0!==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){var Me=(function(e){0,t.inheritsLoose}(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

```

1890 c=(function(){if(c
1891 return!Ember.test
1892 u.call(e,c)
1893 return m}
1894 e.cancelPolls
1895 e.queuedPollTasks
1896 var a=new WeakMap
1897 var o,r=0
1898 var i=Object.create
1899 e.queuedPollTasks
1900 function s(e,t){v
1901 if("number"==type
1902 else{var o=a.get(
1903 n=t
1904 void 0!=o&&o.del
1905 define("ember-lif
1906 Object.defineProperty
1907 e._setRegisteredTimers=function(e){v=e;
1908 e.runTask=function(t){var o=arguments.length>2&&void 0!==(arguments[2]&&arguments[2]:0
1909 if(e.isDestroying)return a
1910 var r=(0,n.default)(e,t,"runTask"),s(i(e),l=Ember.run.later((function(){s.delete(1
1911 r.call(e))),o)
1912 s.add(1)
1913 s.add(1)

```

The top screenshot shows the initial state of the page with various DOM elements and the 'document' object. The bottom screenshot shows the state after a 'Property set failed' error, indicating a failure to set a property on an object.

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

125. Defendant has directly infringed, and continues to directly infringe, the claims of the '135 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

126. LinkedIn has had at least constructive notice of the '135 Patent since at least its issuance. LinkedIn will have been on actual notice of the '135 Patent since, at the latest, the service of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the '135 Patent.

127. EBT believes and contends that, at minimum, LinkedIn's knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the '135 Patent since receiving notice of its infringement of the '135 Patent, is necessarily willful, wanton, malicious,

in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the ‘135 Patent.

COUNT IV – INFRINGEMENT OF U.S. PATENT NO. 8,402,179

128. The ‘179 Patent is entitled “Event Notification System and Method.” The application for the ‘179 Patent was filed on July 20, 2012. The patent issued on March 19, 2013. The ‘179 Patent is a continuation of the ‘135 Patent and claims priority to provisional and non-provisional filings dated as far back as December 9, 2005.

129. Plaintiff is the owner of the ‘179 Patent and has all substantial rights to the ‘179 Patent, including the right and standing to sue and recover damages for past, present and future infringement of the ‘179 patent.

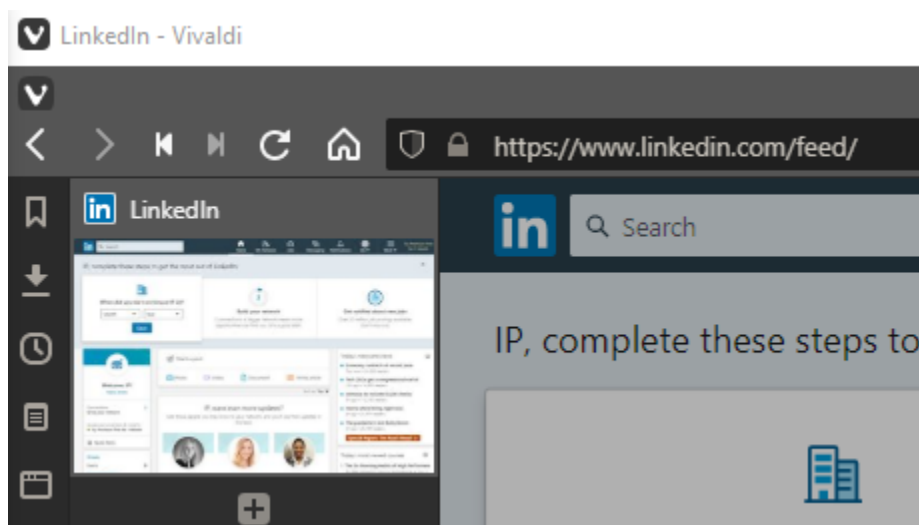
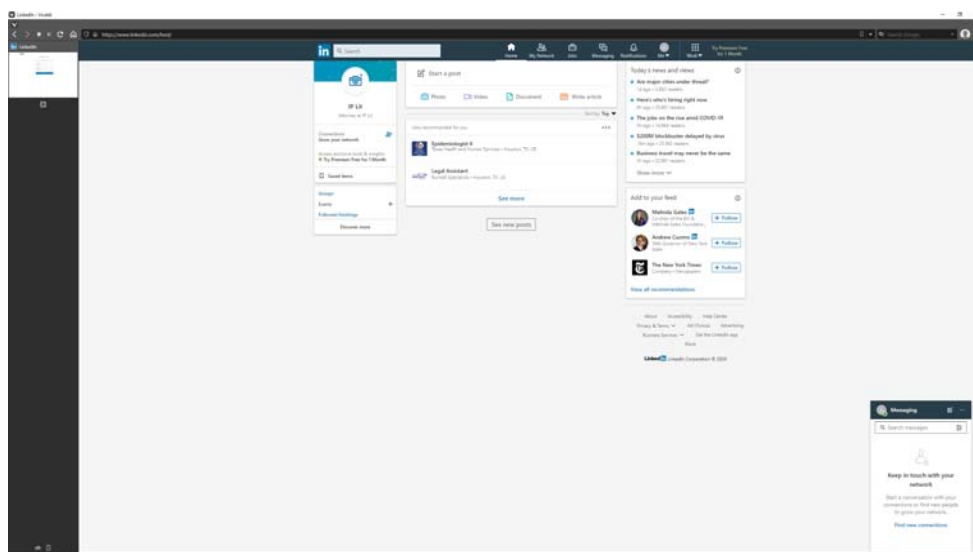
130. Claim 1 of the ‘179 Patent covers a method comprising “processing an event that calls for user notification; generating an event notification for the event; storing the event notification in an array; providing the event notification from the array to a process executed by a processor; using the event notification as a title in association with the process; providing an alternative title from the array to the process; using the alternative title as a title in association with the process.”

131. Defendant has infringed, and is now infringing, the ‘179 Patent, including at least claim 1, in this judicial district and elsewhere, in violation of 35 U.S.C. § 271 through actions comprising the practicing of methods and/or providing of systems, without authority from Plaintiff, for notifying a user of the occurrence of events and notification apparatuses and functionality, including notifying a user of the occurrence of an event by modification of the user’s browser title bar based on the specific event that has occurred, including as claimed in the ‘179

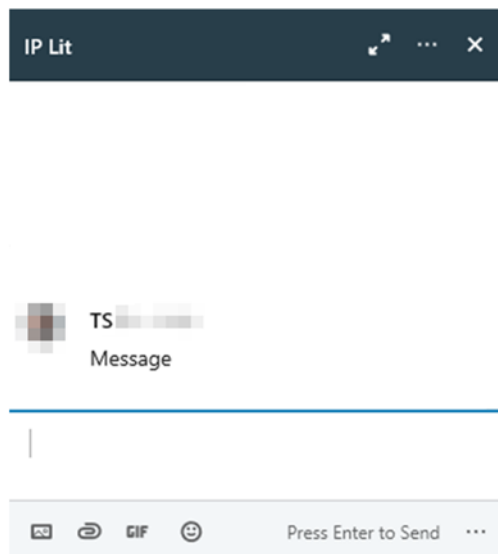
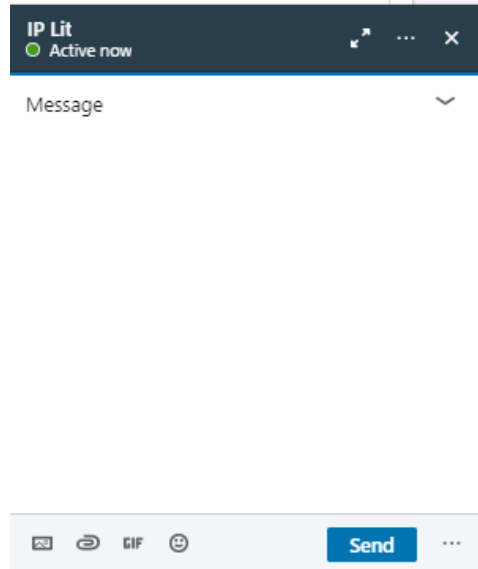
asserted claims. On information and belief, Defendant practices the claimed methods and provides the claimed systems with and via its LinkedIn Application system comprising www.linkedin.com.

132. Without limitation, the accused instrumentality comprising the LinkedIn Application system that practices said systems and methods comprises processing an event that calls for user notification; generating an event notification for the event; storing the event notification in an array; providing the event notification from the array to a process executed by a processor; using the event notification as a title in association with the process; providing an alternative title from the array to the process; using the alternative title as a title in association with the process. For example, the LinkedIn Application system permits a user's device to notify the user, via the user's browser, that the user has notifications for an event, such as a new message and/or other posting on the LinkedIn website, including via updating, modifying, and/or otherwise altering the title bar of the user's browser, including instructing the user's browser to receive notifications regarding messages the user has received from other LinkedIn users and/or posting of other LinkedIn users.

133. The LinkedIn Application processes an event that calls for user notification. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which processes code or other instructions received from LinkedIn that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed/>



See, e.g., Second LinkedIn User Feed page at <https://www.linkedin.com/feed>

```

2114 n=Se(n,r[i])return nfunction Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split(",");
2115 var a=Pe(e,i) a = {value: "/voyager/api/messaging/conversations/6694727273901n=4JGDp_RcmK3ip_RcLnVybjpsaTpmYjYalM6
2116 if(null!=a) return Ne(a,o,n) o = "/voyager/api/messaging/conversations/6694727273901359104/events?q=syncToken&re
2117 if(!r)throw new 1.default("Property set failed: object in path '"+i.join(",")+"' could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "/voyager/api/messaging/conversations/6694727
2119 if(void 0!==(s&&q).has(s)){e[t]-r=n = "/voyager/api/messaging/conversations?keyVersion=LEGACY_INBOX&q=syncToken&s
2120 return n } if(void 0!==(o=e[t])||"object"!=typeof e[t] in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=s&&!(e,t) else e.setUnknownProperty(t,n)

```

```

1835 e.isBlank=He
1836 e.isPresent=function(e){return!He(e)}
1837 e.beginPropertyChanges=ce
1838 e.changeProperties=he
1839 e.endPropertyChanges=fe
1840 e.notifyPropertyChange=le
1841 e.defineProperty=te
1842 e.isElementDescriptor=U
1843 e.nativeDescDecorator=V
1844 e.descriptorForDecorator=D
1845 e.descriptorForProperty=I
1846 e.isClassicDecorator=B
1847 e.setClassicDecorator=F
1848 e.getChainTagsForKey=we
1849 e.getProperties=function(e,t){var n={},r=arguments,i=1
1850 if(2===arguments.length&&Array.isArray(t))i=0
1851 r=arguments[1]for(i<r.length;i++)n[r[i]]=Se(e,r[i])
1852 return n}
1853 e.setProperties=function(e,t){if(null===t||"object"!==typeof t)return t
1854 he((function(){for(var n,r=Object.keys(t),i=0;i<r.length;i++){n=r[i] n = "_lastEventId", r = (2) ["_lastConversatio
1855 Ne(e,n,t[n])}}))
1856 return t}
1857 e.expandProperties=Oe

```

Array(2)

```

0: "_lastConversationId"
1: "_lastEventId"
[]: (...)
firstObject: (...)
hasArrayObservers: (...)
lastObject: (...)
length: 2
__proto__: Array(0)

```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
24000 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
24001 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
24002 this._titles=this._titles.concat(e)}
24003 if(!this._pollerEnabled){this._pollerEnabled=10
24004 this.pollTask("loopTitles","",concat("document-title-poller","").concat(Date.now()))},resetTitle:function(){if(t
24005 this._loopNumber=0
24006 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
24007 this._pollerEnabled=1},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
24008 this._titles[n-1]=this._defaultDocumentTitle
24009 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
24010 t.default&&a&&(document.title=a)
24011 this._loopNumber++
24012 this._pollerEnabled&&this.runTask(e,1500)}}
24013 e.default=n}
24014 define(["image-edit-base/components/hq-image-edit"] ["exports" "image-edit-base/components/picture-cropper" "image-edit

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 if(null!=a) return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)("latestEvent") l1===a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 if(void 0!==(o=s&&q.has(s))){o=t=n=u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 return n}if(void 0!==(o=e[t])||"object"!==typeof e||t in e||"function"!==typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)

```

```

2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {value: u, writable: true, enumerable: true, configurable: true}
2116 if(null!=a) return Ne(a,o,n) o = undefined, n = u {collectionName: "6694725875751120896", spam: false, store: n,
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)("event") s=null===a?void 0:a.set o = undefined, a = {value: u, writable: true, enu
2119 if(void 0!==(o=s&&q.has(s))){o=t=n=u {collectionName: "6694725875751120896", spam: false, store: n, clientSens
2120 return n}if(void 0!==(o=e[t])||"object"!==typeof e||t in e||"function"!==typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)

```

```

2112 void 0==(n=e[t])&&(o=t in e||"function"!==typeof e.setUnknownProperty){n=e.setUnknownProperty(t,n)
2113 if(a&&ke((Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==(o=s&&q.has(s))){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!==typeof e||t in e||"function"!==typeof e.setUnknownProperty){e[t]=n
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=function(e){(!0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a="function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(o=t in e||"function"!==typeof e.setUnknownProperty){n=e.setUnknownProperty(t,n)
2113 if(a&&ke((Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.')}}(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==(o=s&&q.has(s))){e[t]=n n = "LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!==typeof e||t in e||"function"!==typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!==(n&&le(e,t))else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=function(e){(!0,t.inheritsLoose)(i,e)
2123 function i(e){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

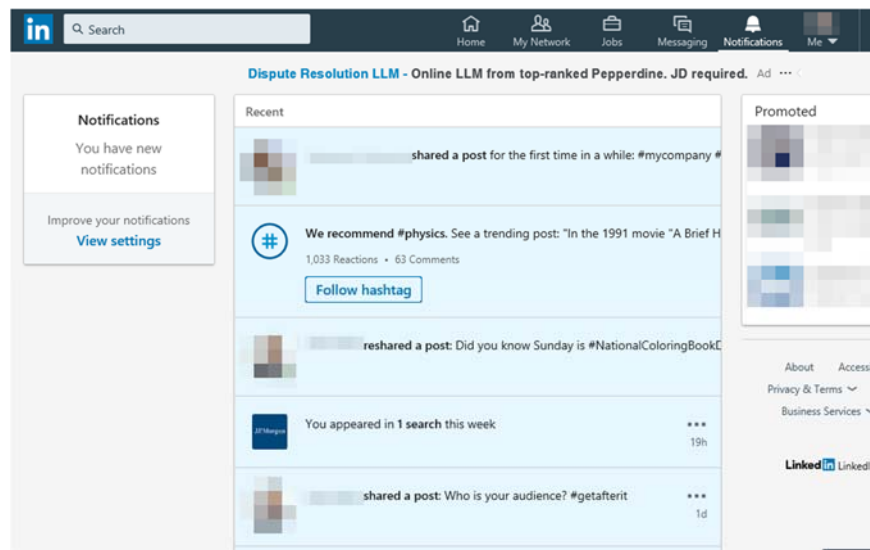
```

```

MessageEvent
  isTrusted: false
  lastEventId: "null"
  origin: ""
  ▶ path: []
  ▶ ports: []
  returnValue: true
  source: null
  srcElement: null
  target: null
  timeStamp: 6400162.895000001
  type: "message"
  userActivation: null
  ▶ __proto__: MessageEvent

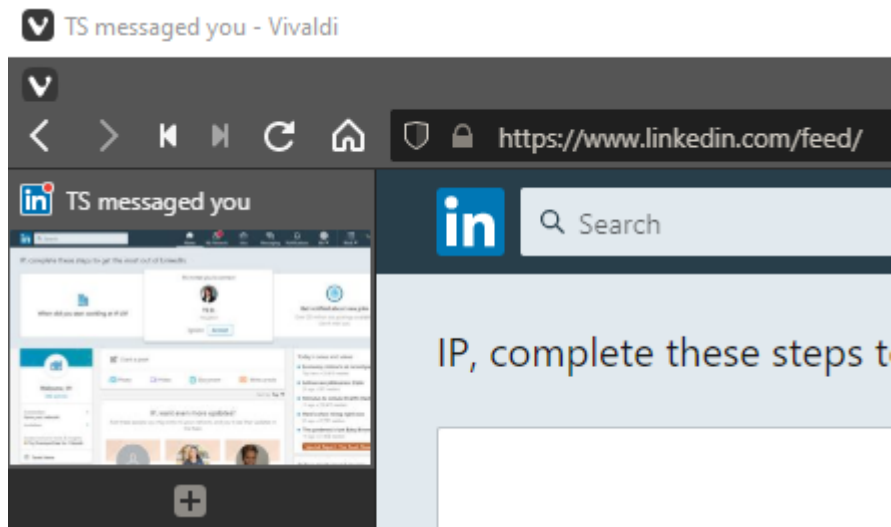
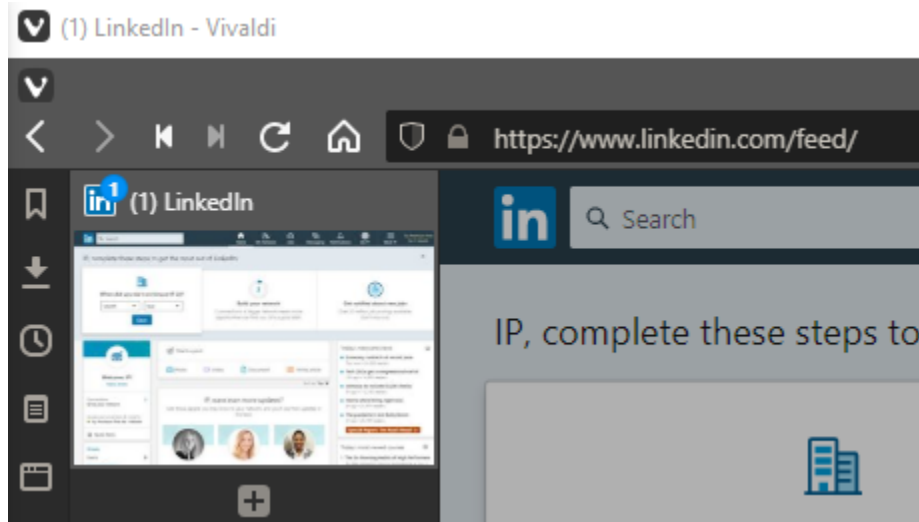
```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>



See, e.g., LinkedIn User Notification page at <https://www.linkedin.com/notifications>

134. The LinkedIn Application generates an event notification for the event. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which generates a notification to indicate to the user that the user has received a message from another LinkedIn user and/or a posting has been made by another LinkedIn user:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if([this._titles.length&t.default][this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if([this._pollerEnabled][this._pollerEnabled=0
23404 this._pollTask(-_loopTitles",-).concat('document-title-poller",-).concat(Date.now()))],resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle)(document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=1)),updateDefaultTitle:function(e){this._defaultDocumentTitle=e,getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var e=this._titles[this._loopNumber] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)))
23413 e.default=n))
23414 define([image-edit-base/components/bg-img-edit","@vovante","image-edit-base/components/picture-cropper","image-edit

```

```

2112 void 0==(n=e[t])&&(!t in e)||function !=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&ee())Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return nfunction Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)●return Ne(a,o,n)o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n;if(void 0!==(o=e[t])||"object"!=typeof e[t]||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o||"function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(!t in e)||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&ee())Z(ie(e,t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))return n;functi
2114 n=Se(n,r[i])return nfunction Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)●return Ne(a,o,n)o = "(1) LinkedIn", n = "LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "'+i.join(".")+" could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "LinkedIn"
2120 return n;if(void 0!==(o=e[t])||"object"!=typeof e[t]||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2124 n._readOnly=!1

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

135. The LinkedIn Application stores the event notification in an array. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings:

```

define("html-document-title/services/document-title-poller",["exports",
  "use-strict",
  Object.defineProperty(e,"__esModule",{
    value:!0
  })
  e.default=void 0
  var n=Ember.Service.extend({
    init:function(){
      this._super.apply(this,arguments)
      this._titles=[]
      this._loopNumber=0
      this._defaultDocumentTitle=""
    },
    addTitle:function(e){
      if(!this._titles.length&&t.default){
        this.updateDefaultTitle(document.title)
        this._titles=[e,this._defaultDocumentTitle]
      }else{
        this._titles.unshift(e)
        this._titles=this._titles.uniq()
        if(!this._pollerEnabled){
          this._pollerEnabled=!0
          this.pollTask("_loopTitles","",concat("docume
        ),

```



```

6415 define("ember-lifeline/poll-task", ["exports", "ember-lifeline/utils/get-task", "ember-lifeline/utils/disposable"],
6416 "use-strict"
6417 Object.defineProperty(exports, "__esModule", {
6418   value: true
6419 })
6420 exports.setRegisteredPollers = function(e) {
6421   e = e
6422 }
6423 exports.setShouldPoll = function(e) {
6424   e = e
6425 }
6426 exports.pollTask = function(e, r) {
6427   var e, n = arguments.length > 2 && void 0 !== arguments[2] ? arguments[2] : 1, u = 0,
6428   t = e.get("pollTask"), d = function() {
6429     return e.call(e, e)
6430   }, n = e.get(e)
6431   if (!n) {
6432     p = new Set
6433     a.set(e, p);
6434     (0,
6435     n.registerDisposable)(e, (function(e, t) {
6436       return function() {
6437         t.forEach(function(t) {
6438           s(e, t)
6439         })
6440       }
6441     )
6442     )(e, p)
6443   }
6444   p.add(a)
6445   e = (function() {
6446     if (a)
6447       return a()
6448     return !ember-testing
6449   })() > d ? function() {
6450     if (a) ~ d
6451   } : d
6452   u.call(e, c)
6453   return m
6454 }
6455 exports.cancelPoll = s
6456 exports.queuedPollTasks = void 0
6457 var a = new WeakMap
6458 var o, r = 0
6459 var i = Object.create(null)
6460 exports.queuedPollTasks = i
6461 function s(e, t) {
6462   var n
6463   if ("number" == typeof e || "string" == typeof e)
6464     n = e
6465   else {
6466     var o = e.get(e)
6467     n = t
6468     void 0 !== o && o.delete(n)
6469   }
6470   delete i[n]
6471 }
6472 function l() {
6473   return r++
6474 }
6475 }
6476 }
6477 })

```

```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("loopTitles","",concat("document-title-poller","").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e,getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500))}
23413 e.default=n))
23414 define("image-edit-base/components/bg-img-edit" ["exports" "image-edit-base/components/picture-cropper" "image-edit

```

```

2112 void 0===n[e(t)]&&0===t in e||function t!=typeof e.setUnknownProperty||(n=e.setUnknownProperty)(t))
2113 if(a&&e())Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray)(n))&&Z(ie(n,"[]"));0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return nfunction Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(!null!=a)return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new Error("Property set failed: object in path '"+i.join(".")+"' could not be found.")(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&a.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n;if(void 0!==(o=e[t])||"object"!=typeof e||t in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 o=n&&ie(e,t)}else e.setUnknownProperty(t,n)
2122 return n}}function xe(){}var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function l(t){var n;(n=e.call(this))||this._volatile=!1
2124 n._readOnly=!1

```

```

e.HOV[
e.HOV[
functi
define
Object
e.defa
var n
e.defa
define
Object
Object
e.defa
var n
this.
this.
this._defaultDocumentTitle=,addTitle:function(e){
this._titles=[e,this._defaultDocumentTitle]}else th
this._titles=this._titles.uniq()

```



```

8600 id(this,t)
8601 Ud("tooltip: must be pased a control",n)
8602 var a=Ld(this,(t.__proto__||Object.getPrototypeOf(t)).apply(this,arguments))
8603 a.player=e
8604 a._control=n
8605 a._titleText=a._control.el().getAttribute("title")
8606 a.setText(r)
8607 a.on(n,"mouseenter",a._onMouseenter)
8608 a.on(n,"mouseleave",a._onMouseleave)
8609 a.on(n,"dispose",function(){return a.dispose()})
8610 return a;Ad(t,[(key:"createEl",value:function(){var e=Ip[this.options._side]||""
8611 return xd(t.prototype.__proto__||Object.getPrototypeOf(t.prototype),"createEl",this).call(this,"div",{className:"vjs-tooltip "+e},{
8612 this._control.el().removeAttribute("title")
8613 this._isShowing=!0})),(key:"hide",value:function(){if(this._isShowing){this.removeClass("vjs-tooltip-active")
8614 this._control.el().setAttribute("title",this._titleText)
8615 this._isShowing=!1})),(key:"setText",value:function(e){var t=this.player.localize(e)
8616 Ep.textContent(this.el(),t)}),(key:"_onMouseenter",value:function(){this.show()}),(key:"_onMouseleave",value:function(){this.hide()})
8617 return t})()
8618 An.prototype.options._=(name:"ControlTooltip")

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

136. The LinkedIn Application provides the event notification from the array to a process executed by a processor. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the entries in the titlebar array for use by the code on the user's browser which is run by the processor of the user's device:

An Introduction to JavaScript

Let's see what's so special about JavaScript, what we can achieve with it, and which other technologies play well with it.

What is JavaScript?

JavaScript was initially created to "make web pages alive".

The programs in this language are called *scripts*. They can be written right in a web page's HTML and run automatically as the page loads.

Scripts are provided and executed as plain text. They don't need special preparation or compilation to run.

In this aspect, JavaScript is very different from another language called *Java*.

Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called the *JavaScript engine*.

The browser has an embedded engine sometimes called a "JavaScript virtual machine".

Different engines have different "codenames". For example:

- V8 – in Chrome and Opera.
- SpiderMonkey – in Firefox.
- ...There are other codenames like "Trident" and "Chakra" for different versions of IE, "ChakraCore" for Microsoft Edge, "Nitro" and "SquirrelFish" for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We'll use them too. For instance, if "a feature X is supported by V8", then it probably works in Chrome and Opera.

How do engines work?

Engines are complicated. But the basics are easy.

1. The engine (embedded if it's a browser) reads ("parses") the script.
2. Then it converts ("compiles") the script to the machine language.
3. And then the machine code runs, pretty fast.

The engine applies optimizations at each step of the process. It even watches the compiled script as it runs, analyzes the data that flows through it, and further optimizes the machine code based on that knowledge.

What can in-browser JavaScript do?

Modern JavaScript is a "safe" programming language. It does not provide low-level access to memory or CPU, because it was initially created for browsers which do not require it.

JavaScript's capabilities greatly depend on the environment it's running in. For instance, [Node.js](#) supports functions that allow JavaScript to read/write arbitrary files, perform network requests, etc.

In-browser JavaScript can do everything related to webpage manipulation, interaction with the user, and the webserver.

For instance, in-browser JavaScript is able to:

- Add new HTML to the page, change the existing content, modify styles.
- React to user actions, run on mouse clicks, pointer movements, key presses.
- Send requests over the network to remote servers, download and upload files (so-called [AJAX](#) and [COMET](#) technologies).
- Get and set cookies, ask questions to the visitor, show messages.
- Remember the data on the client-side ("local storage").

See, e.g., An Introduction to JavaScript on JavaScript.info at <https://JavaScript.info/intro>

137. The LinkedIn Application uses the event notification as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which inputs the created notification into an array of title strings for use by the user's browser to display in the titlebar of the user's browser, including with the HTML title tag processed by the user's browser:

```
23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("_loopTitles","",concat("document-title-poller","_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber%n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}})
23413 e.default=n})
23414 define("image-edit-base/components/bg-img-edit" ["exports", "image-edit-base/components/picture-cropper", "image-edit
```

```
2112 void 0===n[e[t]]&&(t in e) function t=typeof e.unknownProperty||(n=e.unknownProperty(t))
2113 if(a&&e){Z(ie(t));(Array.isArray(n)||(!0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(!0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[1])return n}function Ne(e,t,n){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(!null!=a) return Ne(a,o,n) o = "LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default("Property set failed: object in path '"+i.join(".")+"' could not be found.')(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!==typeof e||t in e||"function"!==typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&ie(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(!0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n._readOnly=!1
```

```

e.HOVI      Array(2)
e.HOVI      0: "TS messaged you"
funct:      1: "(1) LinkedIn"
define:      []: (...)
Object:      firstObject: (...)
e.defi      hasArrayObservers: (...)
e.defi      lastObject: (...)
define:      length: 2
Object:      ► __proto__: Array(0)
Object:
define:
Object:
e.defi
var n:
this:
this:
this._defaultDocumentTitle= ,addTitle:function(e){
● [this._titles=[e,this._defaultDocumentTitle]]else th
this._titles=this._titles.uniq()

```

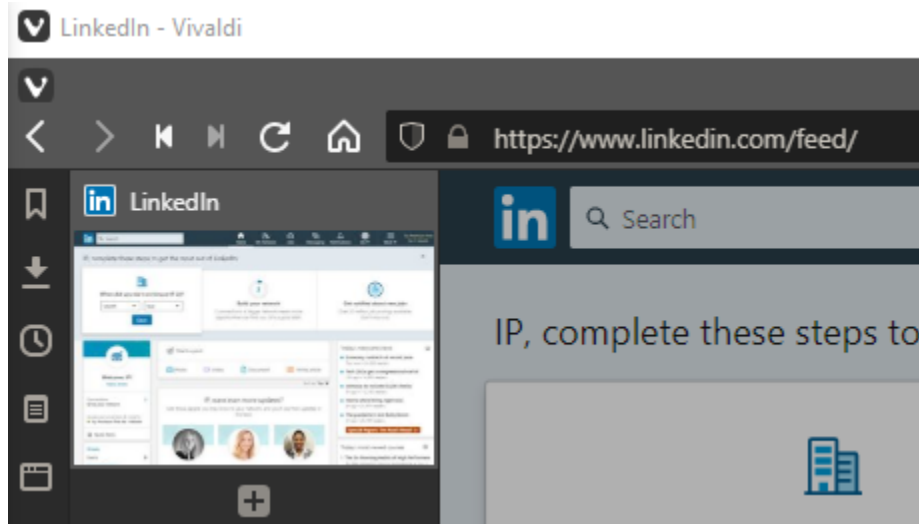
```

8600 Id(this,t)
8601 Ud("tooltip: must be pased a control",n)
8602 var a=Id(this,(t.__proto__||Object.getPrototypeOf(t)).apply(this,arguments))
8603 a.player=e
8604 a._control=n
8605 a._titleText=a._control.el().getAttribute("title")
8606 a.setText(r)
8607 a.on(n,"mouseenter",a._onMouseenter)
8608 a.on(n,"mouseleave",a._onMouseleave)
8609 a.on(n,"dispose",(function(){return a.dispose()}))
8610 return a)Ad(t,[[{key:"createEl",value:function(){var e=Ip[this.options._side]||""
8611 return xd(t.prototype.__proto__||Object.getPrototypeOf(t.prototype),"createEl",this).call(this,"div",{className:"vjs-tooltip "+e},{"
8612 this._control.el().removeAttribute("title")
8613 this._isShowing=!0}}},{key:"hide",value:function(){if(this._isShowing){this.removeClass("vjs-tooltip-active")
8614 this._control.el().setAttribute("title",this._titleText)
8615 this._isShowing=!1}}},{key:"setText",value:function(e){var t=this.player.localize(e)
8616 Ep.textContent(this.el(),t)}},{key:"_onMouseenter",value:function(){this.show()}},{key:"_onMouseleave",value:function(){this.hide()}
8617 return t})()
8618 Ap.prototype.options._name="ControlTooltip"

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

138. The LinkedIn Application provides an alternative title from the array to the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in the titlebar array, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



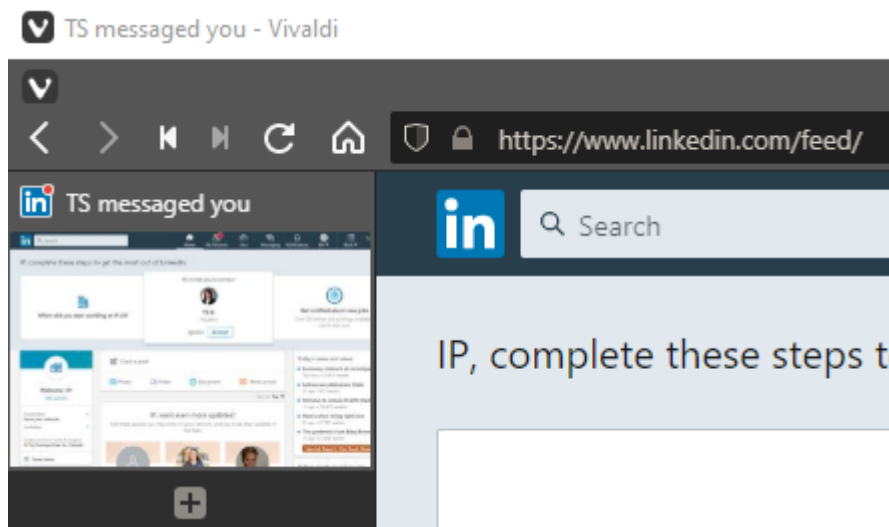
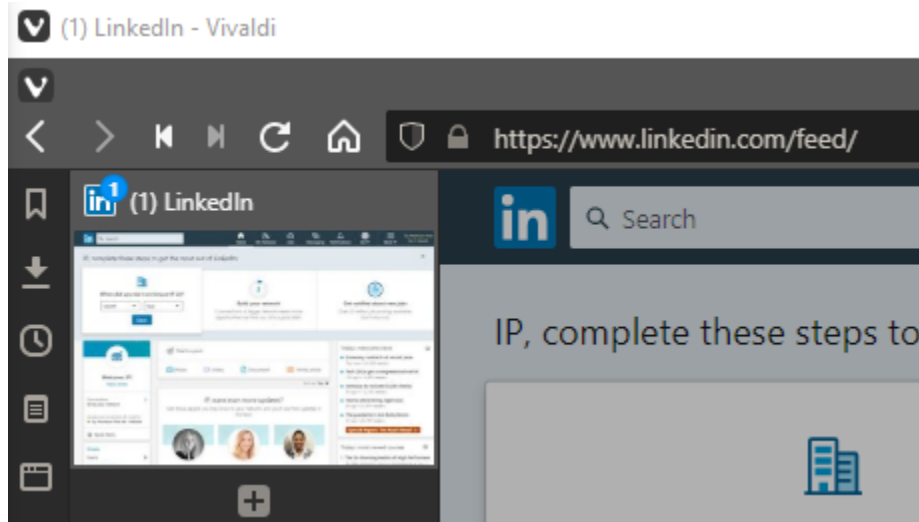
```
23397 var ns=ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle];else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=!0
23404 this.pollTask("._loopTitles","._concat("document-title-poller","._").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=!1}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber&n] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}}
23413 e.default=n}}
23414 define("image-edit-base/components/bg-img-edit" ["exports" "image-edit-base/components/picture-cropper" "image-edit
```

```

2110 void 0==(n[e].c[0]&&[0]||n[e].function!-typeof e.unknownProperty||(n=e.unknownProperty(t))
2111 if(a&&ee(c)){Z(ie(e,t),(Array.isArray(n))||{0,r.isEmberArray}(n))&&Z(ie(n,"[ ]"));{0,r.isProxy}(n)&&Z(ie(n,"content"))
2112 n=Se(n,r[i]);return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split(","),
2113 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2114 • if(null!=a)• return Ne(a,o,n),o = "LinkedIn", n = "(1) LinkedIn"
2115 if(!r)throw new 1.default("Property set failed: object in path '"+i.join(",")+"' could not be found.')}}(e,t,n,i) i
2116 var o,a={0,r.lookupDescriptor}(e,t),s=null===a?void 0:a.set o = "LinkedIn", a = {enumerable: true, configurable: tr
2117 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2118 return n}if(void 0!==(o=e[t])||"object"!-typeof e||t in e||"function"!-typeof e.setUnknownProperty){e[t]=n
2119 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2120 return n}}function xe(){var Me=(function(e){{0,t.inheritsLoose}(i,e)
2121 function i(t){var n;(n=e.call(this)||this)._volatile=!1
2122 n.readOnly=!1
2123

```

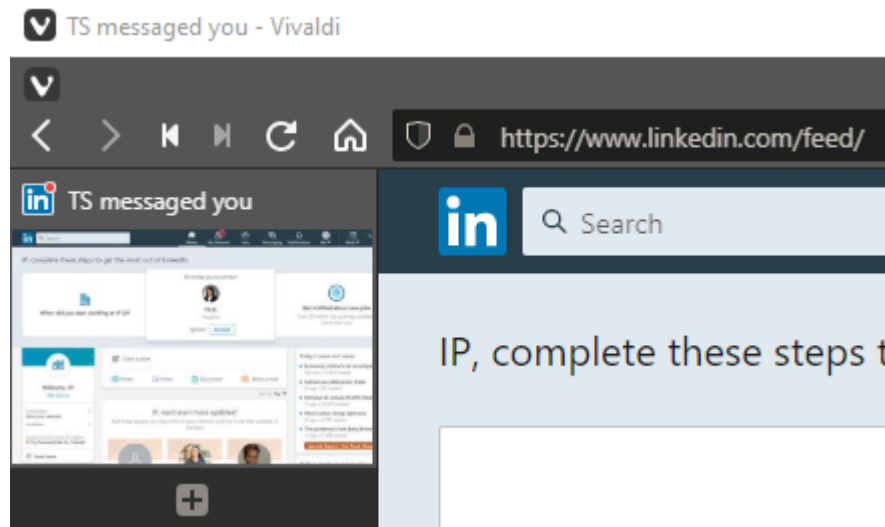
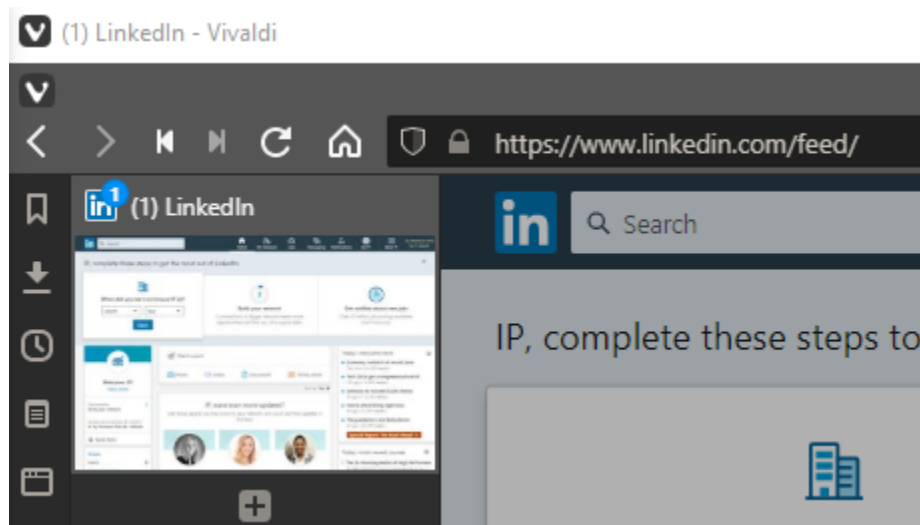
```
e.HOVI Array(2)
e.HOVI 0: "TS messaged you"
funct: 1: "(1) LinkedIn"
defin []: (...)
Object firstObject: (...)
e.defi hasArrayObservers: (...)
var n lastObject: (...)
e.defi length: 2
defin ► __proto__: Array(0)
Object
Object
defin
Object
e.defi
var n
this.
this.
this.
this. defaultDocumentTitle= }, addTitle:function(e){
● this. titles=[e,this. defaultDocumentTitle]}else t
this. titles=this. titles.uniq()
```



See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>

139. The LinkedIn Application uses the alternative title as a title in association with the process. For example, the LinkedIn Application instructs and/or controls the user's browser on the user's device running code which provides the user's browser with titles for each of the entries in

the titlebar array, which is sent to the processor of the user's device for updating the HTML run and displayed by the user's browser:



```

23397 var n=Ember.Service.extend({init:function(){this._super.apply(this,arguments)
23398 this._titles=[]
23399 this._loopNumber=0
23400 this._defaultDocumentTitle="",addTitle:function(e){if(!this._titles.length&&t.default){this.updateDefaultTitle(doc
23401 this._titles=[e,this._defaultDocumentTitle]}else this._titles.unshift(e)
23402 this._titles=this._titles.uniq()
23403 if(!this._pollerEnabled){this._pollerEnabled=10
23404 this.pollTask(" _loopTitles", "",.concat("document-title-poller", "_").concat(Date.now()))},resetTitle:function(){if(t
23405 this._loopNumber=0
23406 if(t.default&&this._defaultDocumentTitle){document.title=this._defaultDocumentTitle
23407 this._pollerEnabled=11}},updateDefaultTitle:function(e){this._defaultDocumentTitle=e},getDefaultTitle:function(){r
23408 this._titles[n-1]=this._defaultDocumentTitle
23409 var a=this._titles[this._loopNumber] a = "TS messaged you"
23410 t.default&&a&&(document.title=a)
23411 this._loopNumber++
23412 this._pollerEnabled&&this.runTask(e,1500)}}
23413 e.default=n))
23414 define("image-edit-base/components/bg-img-edit" ["exports" "image-edit-base/components/picture-cropper" "image-edit-

```

```

2112 void 0==(n=e[t])&&(!o||t in e||function t!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&e())Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray)(n)&&Z(ie(n,"[]"));0,r.isProxy)(n)&&Z(ie(n,"content"))
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: tr
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!typeof e||t in e||"function"!typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n._readOnly=1

```

```

2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o||"function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(!o||t in e||function t!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&e())Z(ie(e,t));(Array.isArray(n)||0,r.isEmberArray)(n)&&Z(ie(n,"[]"));0,r.isProxy)(n)&&Z(ie(n,"content"))}return n}functi
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a) return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new 1.default('Property set failed: object in path "' + i.join(".") + '" could not be found.'))(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a?void 0:a.set o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!typeof e||t in e||"function"!typeof e.setUnknownProperty){e[t]=n o = "(1) LinkedIn"
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)
2122 return n}function xe(){var Me=(function(e){(0,t.inheritsLoose)(i,e)
2123 function i(t){var n;(n=e.call(this))||this._volatile=!1
2124 n._readOnly=1

```

```

1890 c=(function(){if(n
1891 return!Ember.test
1892 u.call(e,c)
1893 return m}
1894 e.cancelPoll=s
1895 e.queuedPollTasks
1896 var a=new WeakMap
1897 var o,r=0
1898 var i=Object.create
1899 e.queuedPollTasks
1900 function s(e,t){v
1901 if("number"==type
1902 else{var o=a.get(e
1903 n=t
1904 void 0!==(o&&o.dele
1905 define("ember-lif
1906 Object.defineProperty
1907 e._setRegisteredTh
1908 e.runTask=function(t){var o=arguments.length>2&&void 0!==arguments[2]?arguments[2]:0
1909 if(e.isDestroying)return a
1910 var r=(0,n.default)(e,t,"runTask"),s=i(e),l=Ember.run.later((function(){s.delete(1)
1911 r.call(e)),o)
1912 s.add(1)
1913 s.add(1)

```

```

2101 u||1==c?Object.defineProperty(e,t,{configurable:!0,enumerable:r.writable:!0,value:l}):e[t]=i}else{l=r
2102 Object.defineProperty(e,t,r)}o.isPrototypeMeta
2103 t.lastRevision=(0,s.value)(t.tag)))
2104 A.has(e)&&A.get(e).forEach((function(t){t.tag=
2105 t.lastRevision=(0,s.value)(t.tag)})))(e)
2106 "function"==typeof e.didDefineProperty&&e.didD
2107 function Ae(e){return"string"==typeof e&&1!==
2108 e.PROXY_CONTENT=Ce
2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0==(n=e[t])&&(!o||t in e||function t!=ty
2113 if(a&&e())Z(ie(e,t));(Array.isArray(n)||0,r
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if
2115 var a=Pe(e,i) a = {enumerable: true, configur
2116 if(null!=a) return Ne(a,o,n) o = "(1) Link
2117 if(!r)throw new 1.default('Property set fail
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null==a
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) Link
2120 return n}if(void 0!==(o=e[t])||"object"!typeof e||t in e||"function"!typeof e.setUnknownProperty){e[t]=n
2121 o!=n&&le(e,t)}else e.setUnknownProperty(t,n)

```



```

2101 u||l1==c?Object.defineProperty(e,t,{configurable:!0,enumerable:c,writable:!0,value:l}):e[t]=i}else{l=r
2102 Object.defineProperty(e,t,r)}o.isPrototypeOfMeta(e)||((function(e){C.has(e)&&C.get(e).forEach((function(t){t.tag=(0,s.combine)(we(e,t.p
2103 t.lastRevision=(0,s.value)(t.tag)})))
2104 A.has(e)&&A.get(e).forEach((function(t){t.tag=(0,s.combine)(we(e,t.path))
2105 t.lastRevision=(0,s.value)(t.tag)})))e
2106 "function"==typeof e.didDefineProperty&&e.didDefineProperty(e,t,l)}var ke=new r.Cache(1e3,function(e){return e.indexOf(".")})
2107 function Ae(e){return"string"==typeof e&&-1==ke.get(e)}var Ce=(0,r.symbol)("PROXY_CONTENT")
2108 e.PROXY_CONTENT=Ce
2109 0
2110 function Se(e,t){var n,i=typeof e,o="object"===i,a=o||"function"===i
2111 if(Ae(t))return a?Pe(e,t):void 0
2112 void 0===n=e[t]&&(!o||t in e||"function"!=typeof e.unknownProperty||(n=e.unknownProperty(t)))
2113 if(a&&ee()){Z(ie(e,t));(Array.isArray(n)||((0,r.isEmberArray)(n))&&Z(ie(n,"[]"));(0,r.isProxy)(n)&&Z(ie(n,"content")))}return n}functi
2114 n=Se(n,r[i])return n}function Ne(e,t,n,i){if(!e.isDestroyed){if(Ae(t))return (function(e,t,n,r){var i=t.split("."),o=i.pop() e = i
2115 var a=Pe(e,i) a = {enumerable: true, configurable: true, get: f, set: f}
2116 if(null!=a)return Ne(a,o,n) o = "(1) LinkedIn", n = "(1) LinkedIn"
2117 if(!r)throw new l.default('Property set failed: object in path "' + i.join(".") + '" could not be found.')})(e,t,n,i) i = undefined, e
2118 var o,a=(0,r.lookupDescriptor)(e,t),s=null===a?voi "title" : o = "(1) LinkedIn", a = {enumerable: true, configurable: true, get: f,
2119 if(void 0!==s&&q.has(s)){e[t]=n n = "(1) LinkedIn"
2120 return n}if(void 0!==(o=e[t])||"object"!=typeof e||!0 in e||"function"!=typeof e.setUnknownProperty){e[t]=n
2121 n=Se(e,t)}else e.setUnknownProperty(t,n)

```

See, e.g., LinkedIn User Feed page at <https://www.linkedin.com/feed>.

140. LinkedIn has directly infringed, and continues to directly infringe, the claims of the '179 Patent, including at least those noted above, including by at least making and using the LinkedIn Application in violation of 35 U.S.C. § 271(a).

141. LinkedIn has had at least constructive notice of the '179 Patent since at least its issuance. LinkedIn will have been on actual notice of the '179 Patent since, at the latest, the service of this complaint. By the time of trial, LinkedIn will have known and intended (since receiving such notice) that its continued actions would actively induce the infringement of the asserted claims of the '179 Patent.

142. EBT believes and contends that, at minimum, LinkedIn's knowing and intentional post-suit continuance of its unjustified, clear, and inexcusable infringement of the '179 Patent since receiving notice of its infringement of the '179 Patent, is necessarily willful, wanton, malicious, in bad-faith, deliberate, conscious and wrongful, and it constitutes egregious conduct worthy of a finding of willful infringement. Accordingly, since at least receiving notice of this suit, LinkedIn has willfully infringed the '179 Patent.

DAMAGES

143. By way of its infringing activities, LinkedIn has caused, and continues to cause, Plaintiff to suffer damages, and Plaintiff is entitled to recover from LinkedIn the damages sustained

by Plaintiff as a result of LinkedIn's wrongful acts in an amount subject to proof at trial, which, by law, cannot be less than a reasonable royalty, together with interest and costs as fixed by this Court under 35 U.S.C. § 284.

144. LinkedIn's infringement of Plaintiff's rights under the Patents-in-Suit will continue to damage Plaintiff, causing irreparable harm for which there is no adequate remedy at law, unless enjoined by this Court.

145. Plaintiff also requests that the Court make a finding that this is an exceptional case entitling Plaintiff to recover its attorneys' fees and costs pursuant to 35 U.S.C. § 285.

DEMAND FOR JURY TRIAL

Pursuant to Rule 38 of the Federal Rules of Civil Procedure, Plaintiff hereby respectfully requests a trial by jury of any issues so triable by right.

PRAYER FOR RELIEF

WHEREFORE, EBT hereby respectfully requests that this Court enter judgment in favor of EBT and against Defendant, and that the Court grant EBT the following relief:

- A. Judgment that Defendant has infringed and is infringing the Patents-in-Suit;
- B. Judgment that LinkedIn's post-notice infringement has been, and continues to be, willful, including that LinkedIn acted to infringe the Patents-in-Suit despite an objectively high likelihood that its actions constituted infringement of a valid patent and, accordingly, award enhanced damages, including treble damages, pursuant to 35 U.S.C. § 284;
- C. An award to Plaintiff of damages adequate to compensate Plaintiff for LinkedIn's infringement, together with pre-judgment and post-judgment interest, and any continuing or future infringement through the date such judgment is entered, including interest, costs, expenses, and an accounting of all infringing acts including, but not limited to, those acts not presented at trial;
- D. A grant of a permanent injunction pursuant to 35 U.S.C. § 283, enjoining LinkedIn and all

persons, including its officers, directors, agents, servants, affiliates, employees, divisions, branches, subsidiaries, parents, and all others acting in active concert or participation therewith, from making, using, offering to sell, or selling in the United States or importing into the United States any methods, systems, or computer readable media that directly or indirectly infringe any claim of the Patents-in-Suit, or any methods, systems, or computer readable media that are colorably different;

- E. That this Court declare this to be an exceptional case and award Plaintiff reasonable attorneys' fees and costs in accordance with 35 U.S.C. § 285; and
- F. Any and all further relief for which Plaintiff may show itself justly entitled that this Court deems just and proper.

Dated: November 6, 2020

Respectfully submitted,

Of Counsel:

FARNAN LLP

Steve Schlather
EDMONDS & SCHLATHER PLLC
2501 Saltus St.
Houston, Texas 77003
Phone: (713) 234-0044
Fax: (713) 224-6651
sschlather@ip-lit.com

/s/ Brian E. Farnan
Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
919 N. Market Street, 12th Floor
Wilmington, DE 19801
Telephone: (302) 777-0300
Fax: (302) 777-0301
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Attorneys for Plaintiff